

2161  
#2

862.C2193

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE



Inventor Application of:

ATSUSHI MIZUNO

Application No.: 09/826,938

Filed: April 6, 2001

For: JOB PROCESSING SYSTEM,  
NETWORK SYSTEM, CONTROL  
METHOD, AND STORAGE  
MEDIUM

Examiner: Unassigned

Group Art Unit: 2161

July 23, 2001

RECEIVED  
JUL 26 2001  
Technology Center 2100

Commissioner for Patents  
Washington, D.C. 20231

CLAIM TO PRIORITY

Sir:

Applicant hereby claims priority under the International Convention and all rights to which he is entitled under 35 U.S.C. § 119 based upon the following Japanese

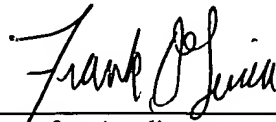
Priority Application:

JP 2000-107126, filed April 7, 2000.

A certified copy of the priority document is enclosed.

Applicant's undersigned attorney may be reached in our New York office by telephone at (212) 218-2100. All correspondence should continue to be directed to our address given below.

Respectfully submitted,



Attorney for Applicant

Registration No. 42,476

FITZPATRICK, CELLA, HARPER & SCINTO  
30 Rockefeller Plaza  
New York, New York 10112-3801  
Facsimile: (212) 218-2200

NY\_MAIN 186553 v 1

(translation of the front page of the priority document of  
Japanese Patent Application No. 2000-107126)



PATENT OFFICE  
JAPANESE GOVERNMENT

RECEIVED  
JUL 26 2001  
Technology Center 2100

This is to certify that the annexed is a true copy of the  
following application as filed with this Office.

Date of Application: April 7, 2000

Application Number : Patent Application 2000-107126

Applicant(s) : Canon Kabushiki Kaisha

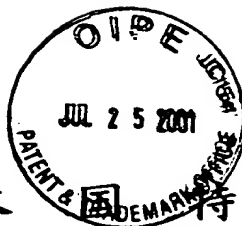
April 27, 2001

Commissioner,

Patent Office

Kouzo OIKAWA

Certification Number 2001-3035626



CFM 2193 US  
09/826,938

日 本 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2000年 4月 7日

出 願 番 号

Application Number:

特願2000-107126

出 願 人

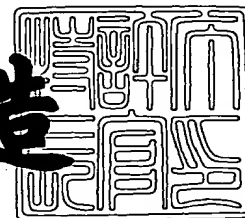
Applicant(s):

キヤノン株式会社

2001年 4月27日

特許庁長官  
Commissioner,  
Japan Patent Office

及川耕造



出証番号 出証特2001-3035626

【書類名】 特許願

【整理番号】 3921013

【提出日】 平成12年 4月 7日

【あて先】 特許庁長官殿

【国際特許分類】 H04N 1/00

【発明の名称】 ジョブ処理システム及びネットワークシステム及び制御  
方法及び記憶媒体

【請求項の数】 12

【発明者】

    【住所又は居所】 東京都大田区下丸子3丁目30番2号 キヤノン株式会  
社内

    【氏名】 水野 敦之

【特許出願人】

    【識別番号】 000001007

    【氏名又は名称】 キヤノン株式会社

【代理人】

    【識別番号】 100076428

    【弁理士】

    【氏名又は名称】 大塚 康徳

    【電話番号】 03-5276-3241

【選任した代理人】

    【識別番号】 100101306

    【弁理士】

    【氏名又は名称】 丸山 幸雄

    【電話番号】 03-5276-3241

【選任した代理人】

    【識別番号】 100115071

    【弁理士】

    【氏名又は名称】 大塚 康弘

【電話番号】 03-5276-3241

【手数料の表示】

【予納台帳番号】 003458

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0001010

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 ジョブ処理システム及びネットワークシステム及び制御方法及び記憶媒体

【特許請求の範囲】

【請求項 1】 同一ネットワークもしくは異なるネットワークどうしが通信可能に接続され、第 1、第 2 の情報処理装置及び出力装置で構成されるジョブ処理システムであって、

前記第 1 の情報処理装置は、

上位処理から渡された出力対象の情報を前記出力装置用のデータにし、  
当該データに対し前記第 2 の情報処理装置からの認証が与えられたときに  
出力開始を行うようにするための属性情報を付加したジョブデータを当該  
出力装置に転送するジョブ発行手段を備え、

前記出力装置は、

受信したジョブデータを記憶保持する記憶保持手段と、  
該記憶保持手段に記憶保持されたジョブデータに対する実行指示情報が  
あった場合に、該当するジョブデータの出力処理を行う制御手段とを備え、  
前記第 2 の情報処理装置は、

前記出力装置に対して前記ジョブデータに対する実行指示情報を与える  
ジョブ開始指示手段を備える  
ことを特徴とするジョブ処理システム。

【請求項 2】 前記第 1 の情報処理装置は、更に、前記ジョブ発行手段で前記ジョブデータを前記出力装置に出力する場合、当該ジョブデータに対する実行を許可するユーザに対してジョブ発行を通知する通知手段を含むことを特徴とする請求項第 1 項に記載のジョブ処理システム。

【請求項 3】 前記通知手段は、前記第 2 の情報処理装置が発行する前記実行指示情報を生成し、その実行指示情報を前記第 2 の情報処理装置に通知することを特徴とする請求項第 2 項に記載のジョブ処理システム。

【請求項 4】 前記第 2 の情報処理装置は、更に、前記通知手段によってジョブ発行の通知を受けた場合に外部に報知する報知手段を備え、

前記ジョブ開始指示手段は、所定の操作が行われた場合、当該ジョブに対する前記実行指示情報を与えることを特徴とする請求項第 2 項に記載のジョブ処理システム。

【請求項 5】 前記第 2 の情報処理装置は、更に、前記ジョブに対する出力許可を与える他のユーザに前記通知手段と同様の情報を通知する手段と、

前記出力装置に対する属性中の出力許可を与えるユーザを追加修正する手段を備えることを特徴とする請求項第 1 項乃至第 4 項に記載のジョブ処理システム。

【請求項 6】 前記第 1 の情報処理装置のジョブ発行手段で発行する属性情報には、ジョブデータに対する出力上限回数が含まれ、

前記出力装置は、前記出力回数上限になった場合に、当該ジョブを抹消する手段を更に備えることを特徴とする請求項第 1 項乃至第 5 項に記載のジョブ処理システム。

【請求項 7】 前記第 1 の情報処理装置のジョブ発行手段で発行する属性情報には、ジョブデータの有効期限に関する情報が含まれ、

前記出力装置は、前記有効期限外のジョブデータを消去する手段を更に備えることを特徴とする請求項第 1 項乃至第 6 項に記載のジョブ処理システム。

【請求項 8】 同一ネットワークもしくは異なるネットワークどうしが通信可能に接続され、第 1、第 2 の情報処理装置及び出力装置で構成されるジョブ処理システムの制御方法であって、

前記第 1 の情報処理装置は、

上位処理から渡された出力対象の情報を前記出力装置用のデータにし、  
当該データに対し前記第 2 の情報処理装置からの認証が与えられたときに  
出力開始を行うようにするための属性情報を付加したジョブデータを当該  
出力装置に転送するジョブ発行工程を備え、

前記出力装置は、

受信したジョブデータを記憶保持する記憶保持工程と、

該記憶保持手段に記憶保持されたジョブデータに対する実行指示情報が  
あった場合に、該当するジョブデータの出力処理を行う制御工程とを備え、  
前記第 2 の情報処理装置は、



前記出力装置に対して前記ジョブデータに対する実行指示情報を与える  
ジョブ開始指示工程を備える  
ことを特徴とするジョブ処理システムの制御方法。

【請求項 9】 請求項第 8 項に記載の各工程に対応するプログラムコードを  
格納することを特徴とする記憶媒体。

【請求項 10】 外部より受信したジョブデータを記憶保持し、当該ジョブ  
データに含まれる属性情報に対して合致する情報を受信した場合に前記ジョブデ  
ータに対する処理を開始する出力装置を有するネットワークシステムであって、  
ネットワーク上に設けられた第 1、第 2 の情報処理装置を有し、  
前記第 1 の情報処理装置は、

上位処理から渡された出力対象の情報を前記出力装置用のデータにし、  
当該データに対し前記第 2 の情報処理装置からの認証が与えられたときに  
出力開始を行うようにするための属性情報を付加したジョブデータを当該  
出力装置に転送するジョブ発行手段を備え、  
前記第 2 の情報処理装置は、

前記出力装置に対して前記ジョブデータに対する実行指示情報を与える  
ジョブ開始指示手段を備える  
ことを特徴とするジョブ処理システム。

【請求項 11】 外部より受信したジョブデータを記憶保持し、当該ジョブ  
データに含まれる属性情報に対して合致する情報を受信した場合に前記ジョブデ  
ータに対する処理を開始する出力装置、及び、第 1、第 2 の情報処理装置を有す  
るネットワークシステムの制御方法であって、  
前記第 1 の情報処理装置は、

上位処理から渡された出力対象の情報を前記出力装置用のデータにし、  
当該データに対し前記第 2 の情報処理装置からの認証が与えられたときに  
出力開始を行うようにするための属性情報を付加したジョブデータを当該  
出力装置に転送するジョブ発行工程を備え、  
前記第 2 の情報処理装置は、

前記出力装置に対して前記ジョブデータに対する実行指示情報を与える

ジョブ開始指示工程を備える

ことを特徴とするネットワークシステムの制御方法。

【請求項 1 2】 請求項第 1 1 項に記載の工程に対応するプログラムコードを格納することを特徴とする記憶媒体。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明はジョブ処理システム及びネットワークシステム及びその制御方法及び記憶媒体に関するものである。

【 0 0 0 2 】

【従来の技術】

近年、コンピュータを相互に接続したローカルエリアネットワーク（LAN）が普及しており、このようなローカルエリアネットワークは、ビルのフロアまたはビル全体、ビル群（構内）、地域、あるいはさらに大きいエリアにわたって構築することができる。このようなネットワークは更に相互に接続され、世界的規模のネットワーク、いわゆる“インターネット”を形成している。

【 0 0 0 3 】

そして、LANもしくはインターネットには、コンピュータの他に、プリンタやファクシミリといったコンピュータ周辺機器も接続されていることが多く、コンピュータからネットワーク経由で利用可能になっている。

【 0 0 0 4 】

このようにネットワークに接続されたコンピュータ周辺機器は、一般に、コンピュータの台数もしくはコンピュータを利用するユーザ数よりも少なく、複数のコンピュータもしくはユーザにより共有されている。

【 0 0 0 5 】

【発明が解決しようとする課題】

ところで、ネットワークの或るコンピュータ上で作成した文書を、他人に渡す場合、一般に次の 2 通りのいずれかで行っていた。

・ 自身が印刷して、その印刷物を手渡す、もしくは郵送する。

・作成した文書の電子ファイル（文書ファイル）を、受取人に転送、もしくは媒体に記憶して渡し、出力してもらう。

【 0 0 0 6 】

しかしながら、前者については即時性に欠ける、出力結果を送る場合には一般的にコストが高くなる、という問題がある。

【 0 0 0 7 】

一方、後者については、その文書ファイルを作成するときに用いたアプリケーションが、相手先の装置にインストールされていなければならない、という問題がある。

【 0 0 0 8 】

上記問題を解決するために、受取人の近くにあるプリンタや F A X ヘデータを送信、出力し、受取人に対して、電話等でその旨を知らせるという手段も考えられるが、この場合に以下の問題が残る。

【 0 0 0 9 】

・受取人の都合によらず、当該コンピュータ周辺機器からデータが出力されてしまうため、出力結果が受取人以外の人に見られる可能性があり、機密性が欠ける。

【 0 0 1 0 】

・同じく受取人の都合によらず、当該コンピュータ周辺機器からデータが出力されてしまうため、当該コンピュータ周辺機器の出力トレイにたくさんの出力用紙が溜まってしまう。

【 0 0 1 1 】

本発明は上記問題点に鑑みなされたものであり、ネットワーク上の第 1 のユーザが第 2 のユーザに対して情報を提供する際、その情報の元になる電子データを閲覧する機能を第 2 のユーザが使用する情報処理装置に無くても、当該第 2 のユーザが視覚可能に出力させることを可能ならしめるジョブ処理システム及びネットワークシステム及びその制御方法及び記憶媒体を提供しようとするものである。

【 0 0 1 2 】

【課題を解決するための手段】

この課題を解決するため、例えば本発明のジョブ処理システムは以下の構成を備える。すなわち、

同一ネットワークもしくは異なるネットワークどうしが通信可能に接続され、第 1、第 2 の情報処理装置及び出力装置で構成されるジョブ処理システムであって、

前記第 1 の情報処理装置は、

上位処理から渡された出力対象の情報を前記出力装置用のデータにし、当該データに対し前記第 2 の情報処理装置からの認証が与えられたときに出力開始を行うようにするための属性情報を付加したジョブデータを当該出力装置に転送するジョブ発行手段を備え、

前記出力装置は、

受信したジョブデータを記憶保持する記憶保持手段と、

該記憶保持手段に記憶保持されたジョブデータに対する実行指示情報があつた場合に、該当するジョブデータの出力処理を行う制御手段とを備え、前記第 2 の情報処理装置は、

前記出力装置に対して前記ジョブデータに対する実行指示情報を与えるジョブ開始指示手段を備える。

【 0 0 1 3 】

【発明の実施の形態】

以下、添付図面に従って本発明にかかる実施形態を詳細に説明する。

【 0 0 1 4 】

<第 1 の実施形態>

まず本発明の実施形態が利用するネットワークについて説明する。なお、以下ではパーソナルコンピュータ（ワークステーションでも良い）を PC と表記することにする。

【 0 0 1 5 】

図 2 は、実施形態が適用するネットワークシステムの例を示している。

【 0 0 1 6 】

図示において、102は開放型アーキテクチャを持つプリンタであり、ネットワークに接続するためのネットワークボード(NB)101を装着している。NB101はローカルエリアネットワーク(LAN)100へ、例えば、同軸コネクタをもつEthernetインターフェース10Base-2や、RJ-45を持つ10Base-T等のLANインターフェースを介してつながれている。

【0017】

PC103やPC104等の複数のパーソナルコンピュータ(PC)もまた、ネットワーク100に接続されており、ネットワークオペレーティングシステムの制御の下、これらのPCはNB101と通信することができる。なお、プリンタには、プリンタ101のように直接ネットワークに接続されているもの、プリンタ105のようにPCに接続されているものがある。

【0018】

通常、LAN100などのLANは、一つの建物内の一つの階又は連続した複数の階でのユーザグループ等の、幾分ローカルなユーザグループにサービスを提供する。例えば、ユーザが他の建物や他県に居るなど、あるユーザが他のユーザから離れるに従って、ワイドエリアネットワーク(WAN)を作ってもよい。WANは、基本的には、いくつかのLANを高速度サービス総合デジタルネットワーク(ISDN)電話線等の高速度デジタルラインで接続して形成された集合体である。

【0019】

従って、図2に示すように、LAN100と、LAN110と、LAN120とは変調/復調(MODEM)/トランスポンダー130及びバックボーン140を介して接続されWANを形成する。これらの接続は、数本のバスによる単純な電氣的接続である。それぞれのLANは専用のPCを含み、また、必ずしも必要なわけではないが、通常はファイルサーバー及びプリントサーバーを含む。

【0020】

従って図2に示すように、LAN110は、PC111と、PC112と、ファイルサーバー113と、ネットワークディスク114と、プリントサーバー115と、プリンタ116及びプリンタ117とを含む。対照的に、LAN120

はPC121とPC122のみを含む。LAN100と、LAN110と、LAN120とに接続されている機器は、WAN接続を介して、他のLANの機器の機能にアクセスすることができる。

【0021】

上記ネットワーク環境において、ネットワーク上のPCが、ネットワーク上の他のデバイスに対してジョブを生成する例を説明する。より具体的には、ネットワーク110上のPC111から、ネットワーク100の或るユーザ向けに、そのネットワーク100上のプリンタ101で印刷出力する場合である。

【0022】

図3は、ジョブ生成ソフトウェアが稼動可能なPCの構成を示すブロック図であり、全PCは全て同様の構成をしているものであるが、便宜的にここではPC111として説明する。

【0023】

PC111は、ROM502もしくはハードディスク（HD）511に記憶された、あるいはフロッピーディスクドライブ（FD）512より供給されるジョブ生成ソフトウェアを実行するCPU501を備え、システムバス504に接続される各デバイスを総括的に制御する。503はRAMで、CPU501の主メモリ、ワークエリア等として機能する。

【0024】

505はキーボードコントローラ（KBC）で、キーボード（KB）509や不図示のポインティングデバイス等からの指示入力を制御する。506はCRTコントローラ（CRTC）で、CRTディスプレイ（CRT）510の表示を制御する。507はディスクコントローラ（DKC）で、ブートプログラム、種々のアプリケーション、編集ファイル、ユーザファイルそしてネットワーク管理プログラム等を記憶するハードディスク（HD）511およびフロッピーディスクコントローラ（FD）512とのアクセスを制御する。508はネットワークインタフェースカード（NIC）で、LAN100を介して、エージェントあるいはネットワーク機器と双方向にデータをやりとりする。

【0025】

先に説明したように、上記構成は P C 1 1 1 だけのものではなく、他の P C も同様の構成を成す。

【 0 0 2 6 】

<<ジョブ生成ソフトウェアの構成>>

次に、本実施形態におけるジョブ生成ソフトウェアの構成について説明する。

【 0 0 2 7 】

本発明のジョブ生成装置は、図 3 に示したようなジョブ生成ソフトウェアを実行可能な P C 上に実現される。ハードディスク ( H D ) 5 1 1 には、後述のすべての説明で動作主体となる本願に係るジョブ生成ソフトウェアのプログラムが格納される。後述のすべての説明において、特に断りのない限り、実行の主体はハード上は C P U 5 0 1 である。一方、ソフトウェア上の制御の主体は、ハードディスク ( H D ) 5 1 1 に格納されたジョブ生成ソフトウェアである。

【 0 0 2 8 】

なお本願に係るジョブ生成ソフトウェアプログラムは、フロッピーディスクや C D - R O M などの記憶媒体に格納された形で供給されても良く、その場合には図 3 に示すフロッピーディスクコントローラ ( F D ) 5 1 2 または不図示の C D - R O M ドライブなどによって記憶媒体からプログラムが読み取られ、ハードディスク ( H D ) 5 1 1 にインストールされる。

【 0 0 2 9 】

図 1 は、上記構成の P C 1 1 1 を、ジョブ生成ソフトウェアおよびその周辺機能部として見た場合の概念構成図である。なお、他の P C も同様の構成であることを付言しておく。

【 0 0 3 0 】

図中、2 は、ジョブデータの発行やジョブデータの出力実行指示を行うジョブ生成ソフトウェアであり、図 3 におけるハードディスク 5 1 1 に格納され、 C P U 5 0 1 によって実行される。またジョブ生成ソフトウェア 2 内部のジョブ認識用メモリ 1 4 やジョブ実行情報用メモリ 1 5 は R A M 5 0 3 が使用され、 R A M 5 0 3 は C P U 5 0 1 がジョブ生成ソフトウェアを実行するためのワークエリアとしても使用される。

【 0 0 3 1 】

3 は、ユーザの入力をジョブ生成ソフトウェア 2 へ伝達する入力部であり、図 3 におけるキーボードコントローラ 5 0 5 で実現される。4 は、ジョブ生成ソフトウェア 2 がネットワークとの通信を行う場合のネットワークプロトコルおよびネットワークボードの制御を行う、ネットワーク制御部であり、CPU 5 0 1 およびNIC 5 0 8 により実現される。

【 0 0 3 2 】

以上のような構成において、本実施形態では、PC 1 1 1 及びPC 1 0 3 上で上記ジョブに関するソフトウェアを起動して、PC 1 1 1 からPC 1 0 3 が接続されたネットワーク上のプリンタ 1 0 2 においてジョブの印刷（出力）を行う。この場合の動作手順について以下にその詳細を説明する。

【 0 0 3 3 】

なお、以下の処理を行う前の前提条件として、プリンタ 1 0 2 では本ジョブ生成ソフトウェアに対応するジョブ出力機能を実装しているものとする（詳細は後述）。

【 0 0 3 4 】

まず、図 1 および図 4 を用いて、本実施形態の動作概要を説明する。本ジョブ生成ソフトウェアは、図 1 に示すように、

- ・ジョブ発行手段 1 0
- ・「ジョブ発行通知」送信手段 1 1
- ・「ジョブ発行通知」受信手段 1 2
- ・ジョブ実行指示手段 1 3
- ・ジョブ認証情報用メモリ 1 4
- ・ジョブ実行情報用メモリ 1 5

によって構成される。

【 0 0 3 5 】

本実施形態では、図 4 に示すように、PC 1 1 1 において、ジョブ発行手段（1 0）と「ジョブ発行通知」送信手段（1 1）が順次実行され、PC 1 0 3 において、「ジョブ発行通知」受信手段（1 2）とジョブ実行指示手段（1 3）が順



次実行される。そして、P C 1 1 1 においては、ユーザ A がジョブ発行人として、ユーザ B へのジョブの発行を実行し、P C 1 0 3 においては、ユーザ B がジョブ受取人として、ジョブの印刷実行を指示する。

【 0 0 3 6 】

では、以下各手段の詳細を説明する。

【 0 0 3 7 】

図 5 のフローチャートは、ジョブ発行手段における処理手順を示している。

【 0 0 3 8 】

このジョブ発行手段は、例えばプリンタドライバに組み込まれていると考えれば分かり易い。すなわち、アプリケーション上で印刷指示したとき、それがトリガになって動作を開始する。

【 0 0 3 9 】

先ず、ステップ S 1 0 1 において、ユーザ A = ジョブ発行人によるジョブ発行指示（使用中のアプリケーション上で）が K B C 5 0 5 から入力される。以下ユーザ A からの入力は K B C 5 0 5 から実行される。

【 0 0 4 0 】

ステップ S 1 0 2 において、ユーザ A はジョブデータの送信先（印刷先）のデバイス（この場合はプリンタ 1 0 2）を識別する情報が入力され、その印刷先デバイス識別情報をジョブ認証情報用メモリ 1 4 へ記録する。印刷先デバイス識別情報としては、印刷先のデバイスの I P アドレスやネットワーク上の名前（D N S がある場合）を使用する。

【 0 0 4 1 】

次いで、ステップ S 1 0 3 において、ユーザ A はジョブの受取人の指定およびジョブ受取人の識別情報を入力し、そのジョブ受取人識別情報を、ジョブ認証情報用メモリ 1 4 に記録する。ジョブ受取人を指定する場合、電子メールのアドレス帳情報や、サーバやユーザグループ等に登録されているネットワークログイン名のリストを使用する。そして、ジョブ受取人識別情報としては、電子メールアドレスやログイン名などを使用する。なおジョブ受取人は複数指定することが可能である。

## 【0042】

ステップS104では、ユーザAからジョブ実行のための任意のパスワードが入力され、そのパスワードをジョブ認証情報用メモリ14に記録する。ステップS105において、発行中の該ジョブを識別するためのジョブIDを生成し、そのジョブIDをジョブ認証情報用メモリ14に記録する。

## 【0043】

ステップS106において、印刷するためのデータ本体と、ジョブ認証情報用メモリ14に記録された上記の各情報、およびその他のジョブ付属属性（ジョブ発行人識別情報やジョブ生成日時など）を基にして、ジョブデータ（プリンタ101が解釈できるプリンタ言語に変換した印刷データを含む）を発行し、ステップS107において、該ジョブデータを、該印刷先デバイスへ送信する。そして、送信された該ジョブデータはプリンタ102内に蓄積される（ステップ106で発行されるジョブデータのデータ構造に関しては図12～図18、およびステップS107で実行される送信シーケンスに関しては、図19に示す）。

## 【0044】

次に、図6のフローチャートにおいて、「ジョブ発行通知」送信手段について説明する。本手段は、前記ジョブ発行手段によって、ジョブが発行されたことを、PC111からPC103へ通知するための「ジョブ発行通知」を送信するための手順である。

## 【0045】

先ず、ステップS201において、ジョブ認証情報用メモリ14から印刷先デバイス識別情報、ジョブID、ジョブ受取人識別情報、およびパスワードを取得する。

## 【0046】

ステップS202において、変数Nにジョブ受取人の数を代入する。ステップS203において、変数N≠0ならばステップS204を実行し、変数N=0ならば本手段を終了する。

## 【0047】

ステップS204において、ジョブ受取人に対して、印刷先デバイス識別情報

、ジョブID、およびパスワードを「ジョブ発行通知」として、ネットワーク経由で送信する。「ジョブ発行通知」をネットワーク経由で送信する具体的な方法に関しては、電子メールもしくはサーバやユーザグループのメッセージングサービスを利用することで実現する。

【0048】

ステップS205において、変数Nを1減らして、ステップS203へ進む。

【0049】

次に、図7のフローチャートにおいて、「ジョブ発行通知」受信手段（PC103側として考えると分かり易い）について説明する。

【0050】

ステップS301において、該「ジョブ発行通知」を受信した旨を、ユーザBに対して表示する。

【0051】

ステップS302において、受信した該「ジョブ発行通知」から、印刷先デバイス識別情報、ジョブID、およびパスワードを取得する。

【0052】

ステップS303において、上記取得した各情報を、ジョブ実行情報用メモリ15へ記録する。

【0053】

次に、図8のフローチャートにおいて、ジョブ実行指示手段（PC103側）について説明する。

【0054】

ステップS401において、ジョブ実行情報用メモリ15から、印刷先デバイス識別情報、ジョブID、およびパスワードを取得する。

【0055】

ステップS402において、上記取得した各情報およびジョブ受取人＝ユーザBの識別情報など基に、ジョブ実行指示コマンドデータを生成する。

【0056】

ステップS403において、印刷先デバイス（プリンタ102）に対し、上記

ジョブ実行指示コマンドデータを送信する。なお、上記ジョブ実行指示コマンドデータのデータ構造に関しては図12、図17、図18、およびその送信シーケンスに関しては、図20に示す。

【0057】

一方、プリンタ102は、例えば図21に示すような構造を成している。

【0058】

図中、200は装置全体の制御を司るCPUであり、202は実施形態における印刷処理手順であるプログラム及びジョブ管理プログラムの他、フォントデータを格納しているROMである。202はCPU200のワークエリアとして使用されるRAMであり、ここには図示の如くジョブ管理テーブル202aが確保されている。203は印刷データをファイルとして格納するためのハードディスクである。101は先に説明したNB、204は各種設定を行うためのスイッチ及びメッセージ等を表示する表示操作パネルを有する操作パネルである。205は印刷出力する際のビットマップイメージを展開する画像メモリ、206はプリンタエンジンインターフェース、207はプリンタエンジンである。

【0059】

図22は、ジョブ管理テーブルの内容を示している。図示の如く、ジョブ管理テーブルは、1つのジョブについて1行が対応しており、ジョブID、受取時間、発行人（上記実施形態ではユーザA）、ユーザAが設定したパスワード、受取人（複数の存在を許容する）、及び、HD203にファイルとして格納された印刷データファイル名で構成される。

【0060】

本プリンタ102は、1つのジョブに対し、複数の受取人が設定可能であるので、受取人の欄には、複数の受取人を特定する情報が格納されると共に、各受取人から印刷指示を受けたときに、その受取人に対して印刷が完了したか否かを示すフラグを設けた。図示の場合、受取人としてAAA、BBB、CCCの3人が存在し、そのいずれからも印刷指示がない状態を示すため“0”をセットするようにした。そして、例えば、AAAというユーザから印刷指示があった場合には、印刷処理を行うと共に、その印刷が正常に完了したときに、そのユーザAAA

に対するフラグを“1”にセットする。そして、全てのユーザのフラグが“1”になった段階で、該当するジョブをジョブ管理テーブル202aから抹消すると共に、HD203に格納されていたファイル（印刷データ）を削除する。

【0061】

上記構成において、本プリンタ102は、ネットワーク上から印刷ジョブを受信情報すると、その中に含まれる情報に基づいてジョブ管理テーブル202aにそのジョブを追加すると共に、実体である印刷データをハードディスク203にファイルとして格納する。

【0062】

そして、ネットワーク上から、ジョブ実行指示（ヘッダを調べることで判定できる）があった場合には、その中に含まれるジョブIDやパスワード、及び要求元情報が記憶されている管理テーブルに存在するか否かを認証処理を行い、この認証処理をパスしたジョブについての印刷を行う。

【0063】

以上の結果、本実施形態によれば、ネットワークを介して、ユーザAからユーザBに対して文書を渡すとき、プリンタより出力される形式のデータを直接プリンタに送信し、ユーザBは単にそのプリンタに対して印刷を行うような作業をするだけで良くなる。従って、受取人であるユーザBが使用するPC上には、ユーザAが使用したアプリケーションがインストールされている必要もなくなり、尚且つ、印刷された書類ではなく電子データ形式で送信するので、低コストで文書のやりとりが可能になる。

【0064】

<第2の実施形態2>

上記第1の実施形態においては、ジョブIDやパスワードなどのジョブ認証およびジョブ実行指示に必要な情報を「ジョブ発行通知」として、ジョブ発行人側＝PC111からジョブ受取人側＝PC103に送信し、ジョブ受取人側にてジョブ実行指示コマンドデータを生成していた。

【0065】

本第2の実施形態においては、ジョブ実行指示コマンドデータそのものをジョ

ブ発行人側＝PC111にて生成する実施形態について記述する。つまり、ジョブ受取人側の処理で、ジョブ実行指示コマンドを生成する処理をなくし、発行人側から送られてきたジョブ実行指示コマンドを単純にプリンタに出力する例である。

【0066】

実行環境にかかるジョブ生成ソフトウェアの構成については、第1の実施形態と同様である。

【0067】

以下、本実施形態における各手段の詳細について説明する。

【0068】

ジョブ発行手段については、第1の実施形態と同様である（図5のフローチャートおよび第1の実施形態を参照）。そこで、先ず、「ジョブ発行通知」送信手段について、図9のフローチャートに従って説明する。

【0069】

ステップS501において、ジョブ認証情報用メモリ14から印刷先デバイス識別情報、ジョブID、ジョブ受取人識別情報、およびパスワードを取得する。

【0070】

ステップS502において、変数Nにジョブ受取人の数を代入する。ステップS503において、変数N≠0ならばステップS504を実行し、変数N=0ならば本手段を終了する。ステップS504において、ジョブ受取人に対して、印刷先デバイス識別情報、ジョブID、およびパスワードを基に、ジョブ実行指示コマンドデータを生成する（ジョブ指示実行コマンドデータのデータ構造の詳細は図12、図17、図18に示す。）

ステップS505において、上記ジョブ実行指示コマンドデータを「ジョブ発行通知」として、ネットワーク経由で送信する。「ジョブ発行通知」をネットワーク経由で送信する具体的な方法に関しては、電子メールもしくはサーバやユーザグループのメッセージングサービスを利用することで実現する。

【0071】

ステップS506において、変数Nを1減らして、ステップS503へ進む。

【 0 0 7 2 】

次に、図 1 0 のフローチャートにおいて、「ジョブ発行通知」受信手段について説明する。

【 0 0 7 3 】

ステップ S 6 0 1 において、該「ジョブ発行通知」を受信した旨を、ユーザ B に対して表示する。ステップ S 6 0 2 において、受信した該「ジョブ発行通知」から、ジョブ実行指示コマンドデータを取得する。ステップ S 6 0 3 において、上記取得したジョブ実行指示コマンドデータを、ジョブ実行情報用メモリ 1 5 へ記録する。

【 0 0 7 4 】

次に、図 1 1 のフローチャートにおいて、ジョブ実行指示手段について説明する。

【 0 0 7 5 】

ステップ S 7 0 1 において、ジョブ実行情報用メモリ 1 5 から、ジョブ実行指示コマンドデータを取得する。ステップ S 7 0 2 において、印刷先デバイスに対し、上記ジョブ実行指示コマンドデータを送信する。ジョブ実行指示コマンドデータの送信シーケンスに関しては、図 2 0 に示す通りである。

【 0 0 7 6 】

以上説明したように本第 2 の実施形態によれば、第 1 の実施形態と同様の作用効果を奏し、しかも受取人側での処理が簡略化されるようになる。

【 0 0 7 7 】

< 第 3 の実施形態 >

上記第 1、第 2 の実施形態では、最終的に印刷出力するのは受取人が印刷指示を行う場合であった。つまり、発行人が一旦、印刷ジョブを発行してしまった後のそのジョブの管理は受取人側次第ということになり、発行人側での介在ができなくなってしまう。また、場合によっては、プリンタ側にジョブが溜まってしまふという問題もある。

【 0 0 7 8 】

そこで、本第 3 の実施形態では、これを解決する例を説明する。

【0079】

説明を簡単にするため、ネットワークの構成は、図2と同じとし、ジョブ発行する側もPC111、受取人側のPCも103、実際に印刷処理するプリンタも102として説明する。

【0080】

<<ジョブ生成ソフトウェアの構成>>

本第3の実施形態におけるジョブ生成ソフトウェアの構成について説明する。

【0081】

図23は、PC111を、ジョブ生成ソフトウェアおよびその周辺機能部として見た場合の、第3の実施形態における概念構成図である。第1の実施形態（図1）と異なる点は、「ジョブ発行通知」の転送手段16を追加した点にある。すなわち、PC111が実装するソフトウェアは、

- ・ジョブ発行手段10
- ・「ジョブ発行通知」送信手段11
- ・「ジョブ発行通知」受信手段12
- ・ジョブ実行指示手段13
- ・ジョブ認証情報用メモリ14
- ・ジョブ実行情報用メモリ15
- ・「ジョブ発行通知」転送手段16

によって構成される。

【0082】

本第3の実施形態では、図4に示すように、PC111において、ジョブ発行手段10と「ジョブ発行通知」送信手段11が順次実行され、PC103において、「ジョブ発行通知」受信手段12が実行された後、ユーザの任意によりジョブ実行指示手段13もしくは「ジョブ発行通知」転送手段16が実行される。そして、PC111においては、ユーザAがジョブ発行人として、ユーザBへのジョブの発行を実行し、PC103においては、ユーザBがジョブ受取人として、ジョブの印刷実行、もしくは、「ジョブ発行通知」の転送を指示する。

【0083】



以下、上記各手段の動作処理内容について詳細を説明する。

【0084】

図5のフローチャートは、ジョブ発行手段における処理手順を示している。

【0085】

先ず、ステップS1101において、ユーザA=ジョブ発行人によるジョブ発行指示（使用中のアプリケーション上）がKBC505から入力される。以下ユーザAからの入力はKBC505から実行される。

【0086】

ステップS1102において、ユーザAはジョブデータの送信先（印刷先）のデバイス（この場合はプリンタ102）を識別する情報が入力され、その印刷先デバイス識別情報をジョブ認証情報用メモリ14へ記録する。印刷先デバイス識別情報としては、印刷先のデバイスのIPアドレスやネットワーク上の名前（DNSがある場合）を使用する。

【0087】

次いで、ステップS1103において、ユーザAはジョブの受取人の指定およびジョブ受取人の識別情報を入力し、そのジョブ受取人識別情報を、ジョブ認証情報用メモリ14に記録する。ジョブ受取人を指定する場合、電子メールのアドレス帳情報や、サーバやユーザグループ等に登録されているネットワークログイン名のリストを使用する。そして、ジョブ受取人識別情報としては、電子メールアドレスやログイン名などを使用する。なおジョブ受取人は複数指定することが可能である。

【0088】

ステップS1104において、前記入力されたジョブ受取人の数を基に、ジョブの実行許可数として算出し、ジョブ認証情報用メモリ14に記録する。そして、ステップS1105において、ユーザAからジョブの有効期限が入力され、そのジョブ有効期限をジョブ認証情報用メモリ14に記録する。ステップS1106では、ユーザAからジョブ実行のための任意のパスワードが入力され、そのパスワードをジョブ認証情報用メモリ14に記録する。

【0089】

ステップ S 1 1 0 7 において、発行中の該ジョブを識別するためのジョブ I D を生成し、そのジョブ I D をジョブ認証情報用メモリ 1 4 に記録する。

【0090】

ステップ S 1 1 0 8 において、印刷するためのデータ本体と、ジョブ認証情報用メモリ 1 4 に記録された上記の各情報、およびその他のジョブ付属属性（ジョブ発行人識別情報やジョブ生成日時など）を基にして、ジョブデータを発行し、ステップ S 1 1 0 9 において、該ジョブデータを、該印刷先デバイス（プリンタ 1 0 2）へ送信する。なお、ステップ S 1 1 0 8 で発行されるジョブデータのデータ構造に関しては、図 2 9、図 1 3～図 1 7、図 3 0、およびステップ S 1 1 0 9 で実行される送信シーケンスに関しては、図 1 9 に示す通りである。

【0091】

そして、送信された該ジョブデータはプリンタ 1 0 2 内に蓄積され、入力されたジョブの有効期限が満了するか、もしくはジョブの実行許可数が 0 になると、プリンタ 1 0 2 から排除（消去）される。

【0092】

本第 3 の実施形態における「ジョブ発行通知」送信手段であるが、これは第 1 の実施形態を同様であるので、その詳細は第 1 の実施形態を参照されたい。

【0093】

次に、図 2 6 のフローチャートを用いて、第 3 の実施形態における「ジョブ発行通知」受信手段（P C 1 0 3 側の処理）について説明する。

【0094】

まず、ステップ S 1 3 0 1 において、「ジョブ発行通知」を受信した旨を、ユーザ B に対して表示する。次いで、ステップ S 1 3 0 2 において、受信した該「ジョブ発行通知」から、印刷先デバイス識別情報、ジョブ I D、ジョブ有効期限およびパスワードを取得する。

【0095】

ステップ S 1 3 0 3 では、上記取得した各情報を、ジョブ実行情報用メモリ 1 5 へ記録する。

【0096】

次に、図 2 7 のフローチャートに従ってジョブ実行指示手段（P C 1 0 3 側）について説明する。

【 0 0 9 7 】

まず、ステップ S 1 4 0 1 において、ジョブ実行情報用メモリ 1 5 から、印刷先デバイス識別情報、ジョブ I D、およびパスワードを取得する。次いで、ステップ S 1 4 0 2 において、上記取得した各情報およびジョブ受取人＝ユーザ B の識別情報など基に、ジョブ実行指示コマンドデータを生成する。そして、ステップ S 1 4 0 3 において、印刷先デバイスに対し、上記ジョブ実行指示コマンドデータを送信する。次いで、ステップ S 1 4 0 4 において、該ジョブの実行許可数のカウンタを 1 減らすため、ジョブ修正属性コマンドデータを生成する。最後に、ステップ S 1 4 0 5 において、印刷先デバイスに対し、上記ジョブ属性修正コマンドデータを送信する。

【 0 0 9 8 】

なお、上記ジョブ実行指示コマンドデータおよびジョブ属性修正コマンドデータのデータ構造に関しては図 2 9、図 1 3 ～図 1 7、及び図 3 0 に示す通りである。また、その送信シーケンスに関しては、図 2 0、図 3 2 に示す通りである。

【 0 0 9 9 】

次に、図 2 8 のフローチャートにおいて、「ジョブ発行通知」転送手段について説明する。本手段は、上記「ジョブ発行通知」受信手段によって、「ジョブ発行通知」が受信された場合に、さらに他のユーザ C へ「ジョブ発行通知」を転送するための手順である。

【 0 1 0 0 】

まず、ステップ S 1 5 0 1 において、ジョブ実行情報用メモリ 1 5 から印刷先デバイス識別情報、ジョブ I D、ジョブの有効期限、およびパスワードを取得する。次いで、ステップ S 1 5 0 2 において、ユーザ B によってジョブ転送先が入力される。

【 0 1 0 1 】

ステップ S 1 5 0 3 では、該ジョブのジョブ受取人を変更するためのジョブ属性修正コマンドデータを生成する。

【 0 1 0 2 】

ステップ S 1 5 0 4 において、該印刷先デバイスに対し、該ジョブ属性修正コマンドデータを送信する。

【 0 1 0 3 】

ステップ S 1 5 0 5 において、ジョブ転送先に対して、印刷先デバイス識別情報、ジョブ ID、およびパスワードを「ジョブ発行通知」として、ネットワーク経由で送信する。「ジョブ発行通知」をネットワーク経由で送信する具体的な方法に関しては、電子メールもしくはサーバやユーザグループのメッセージングサービスを利用することで実現する。

【 0 1 0 4 】

なお、上記ジョブ発行通知の転送を行えるのは、ジョブ発行通知を受けた者のみとなるのは上記処理から理解できよう。

【 0 1 0 5 】

また、上記ジョブ属性修正コマンドデータのデータ構造に関しては、図 2 9、図 3 0、図 3 1 に示し、その送信シーケンスに関しては、図 3 2 に示す通りである。

【 0 1 0 6 】

一方、プリンタ 1 0 2 は、図 2 1 の構成に加えて、内部にタイマを備え計時する機構を備えることになる。そして、ジョブ管理テーブル 2 0 2 a には、ジョブの有効期限、ジョブ実行回数を計数するカウンタ変数の欄が追加されることになる。

【 0 1 0 7 】

そして、現在の日時をタイマより得、各ジョブの有効期限とを比較し、有効期限を過ぎているジョブについてはメモリより抹消する。また、ジョブ実行コマンドを受信した場合には、そのコマンドに含まれるデータに合致するジョブを読み込み実行する（該当する印刷データに基づく印刷処理を行う）。

【 0 1 0 8 】

また、ジョブ属性の修正コマンドがあった場合には、そのコマンドに含まれる修正箇所に対応するテーブル内の該当するレコードを修正する。例えば、ジョブ

実行回数を減らす指示の場合には該当するジョブのカウンタ変数を“1”だけ減らし、それが“0”になったらそのジョブを削除する。また、ジョブの受取人の変更を示すコマンドの場合、ジョブ管理テーブル中の受取人のレコード中、その変更を指示してきた受取人を抹消すると共に、変更指示を受けた新たな受取人を登録する処理を行うことになる。

## 【0109】

具体的な処理例を示せば、図33のようになる。

## 【0110】

先ず、ステップS2001でネットワークより情報が受信されたか否かを判断する。否の場合には、ステップS2002に進み、タイマの時刻と管理テーブル202aに登録された各ジョブの有効期限とを比較し、有効期限を過ぎたジョブがあるかどうかを判断する。有効期限切れのジョブがあった場合には、ステップS2003でそのジョブを管理テーブル202aより削除すると共に、印刷データファイルをHD203より削除する。

## 【0111】

また、ステップS2001で何らかの情報を受信したと判断した場合には、ステップS2004に進み、その情報はジョブ発行データであると判断した場合には、ステップS2005においてそのジョブを受信すると共に印刷データの部分をファイルとしてHD203に格納する。次いで、ステップS2006においてジョブ管理テーブル202aに登録する。

## 【0112】

また、ジョブ発行データではないと判断した場合には、ステップS2007に進んで、ジョブ実行コマンドか否かを判断する。ジョブ実行コマンドであると判断した場合には、そのコマンドで指定されたジョブIDで示されるジョブの受取人の欄に、そのジョブ実行コマンドの送信元（上記実施形態ではPC103）が含まれ、尚且つ、パスワードも合致するか否かを判断することで、認証処理を行う。この認証処理がパスすると、ステップS2009において、該当するジョブを印刷キューに登録する。

## 【0113】

説明が前後するが、実施形態におけるプリンタ102は、ジョブが印刷キューに登録されると、その登録順にそのジョブに対する印刷処理を行う処理プログラムが別途動作しているものとしている。

## 【0114】

さて、ジョブ実行コマンドでもない判断された場合、処理はステップS2007からステップS2010に進み、ジョブ修正コマンドかどうかを判断する。ジョブ修正コマンドには、先に説明したように、カウンタをデクリメントする、受取人を修正する等の処理が含まれる。修正コマンド以外であると判断した場合には、ステップS2011に進み、対応する処理を行うが、修正コマンドであると判断した場合には、ステップS2012で修正対象のジョブに対して、修正コマンドを発行元のユーザがその権利を有するか否かを、ステップS2008と同様にして判断する。次いで、ステップS2014に進み、ジョブ管理テーブル202aを調べ、その中にカウンタが“0”になったジョブがあるか否かを判断し、もし有れば、ステップS2015でそのジョブを消去する。

## 【0115】

以上の結果、本第3の実施形態によれば、上記第1の実施形態に加えて、ジョブの発行元側で、ジョブに対する有効期限が設定できるので、プリンタ内にジョブが無制限に溜まっていくことがなくなり、機密保持に有効に作用することになる。

## 【0116】

また、受取人は、第3者に対してジョブ実行コマンドの発生権利を与えることもできるようにもなる。

## 【0117】

なお、上記第1乃至第3の実施形態で説明したプリンタ102は、上記の処理を実現するための処理プログラムを内蔵するものとして説明したが、例えば、図2におけるプリンタ105が一般的なプリンタであり、それが接続されているPC104がプリントサーバとして機能し、そのプリントサーバのプログラムがプリンタ102と同様の処理を行うようにしても構わない。

## 【0118】

また、上記第 1 乃至第 3 の実施形態では、出力対象デバイスとしてプリンタを用いたが、ネットワーク上に接続された F A X が有するプリンタ部や複写機が有するプリンタ部等でも良いのは勿論である。

## 【 0 1 1 9 】

また、上記実施形態は、ネットワーク、ネットワークへの接続するためのカードもしくはボード等のハードウェアを必要とするものの、パーソナルコンピュータをはじめとする汎用情報処理装置に、組み込むソフトウェアによって実現できる。

## 【 0 1 2 0 】

従って、本発明は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体（または記録媒体）を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（または C P U や M P U ）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム（ O S ）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

## 【 0 1 2 1 】

さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張カードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張カードや機能拡張ユニットに備わる C P U などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

## 【 0 1 2 2 】

以上説明したように、本実施形態によれば、ジョブデータを発行する場合に、ジョブの実行許可に必要なジョブ属性（特に「ジョブ受取人」の識別情報やパスワード）をジョブデータに付加して該ジョブデータを発行してプリンタなどのデバイスへ送信し、該ジョブの実行許可に必要なジョブ属性を、「ジョブ発行通知」として「ジョブ発行人」から「ジョブ受取人」に転送し、「ジョブ受取人」がジョブ実行指示する場合には、該ジョブの実行許可に必要なジョブ属性を付加した、ジョブ実行指示コマンドデータを該デバイスへ送信して実行を指示することにより、「ジョブ受取人」だけが任意のタイミングでジョブの実行結果（印刷出力結果など）を得ることが可能となり、目的となる出力結果を「ジョブ受取人」だけに効率よく転送可能なシステムを提供できるという効果がある。

## 【 0 1 2 3 】

また、ジョブデータを発行する場合に、「ジョブの実行許可に必要なジョブ属性（特に「ジョブ受取人」の識別情報やパスワード）」および「ジョブのライフタイムを管理するジョブ属性（ジョブの有効期限など）」をジョブデータに付加して該ジョブデータを発光してプリンタなどのデバイスへ送信し、該「ジョブの実行許可に必要なジョブ属性」および「ジョブのライフタイムを管理するジョブ属性」を、「ジョブ発行通知」として「ジョブ発行人」から「ジョブ受取人」に転送し、「ジョブ受取人」がジョブ実行指示する場合には、該「ジョブの実行許可に必要なジョブ属性」を付加した、ジョブ実行指示コマンドデータを該デバイスへ送信してジョブ実行を指示することにより、「ジョブ受取人」だけが任意のタイミングでジョブの実行結果（印刷出力結果など）を得ることが可能で、かつ、発行されたジョブのライフタイムにジョブ発行人の意志を反映させることが可能となるため、気密性が高く、また該デバイス内におけるジョブの生存期間が明確なシステムを提供できるという効果がある。

## 【 0 1 2 4 】

また、「ジョブ受取人」がさらに第三者に「ジョブ発行通知」を転送することにより、発行されたジョブの可搬性を向上させることが可能となるため、二者間あるいは三者間におけるジョブの伝達効率を向上させるという効果がある。

## 【 0 1 2 5 】



【発明の効果】

以上説明したように本発明によれば、ネットワーク上の第 1 のユーザが第 2 のユーザに対して情報を提供する際、その情報の元になる電子データを閲覧する機能を第 2 のユーザが使用する情報処理装置が無くても、当該第 2 のユーザが視覚可能に出力させることが可能になる。

【図面の簡単な説明】

【図 1】

第 1 の実施形態におけるジョブ生成ソフトウェアおよびその周辺動作環境の概念図である。

【図 2】

実施形態が適用するネットワーク構成例を示す図である。

【図 3】

実施形態におけるパーソナルコンピュータのブロック構成図である。

【図 4】

第 1 の実施形態におけるジョブ生成ソフトウェアの動作関係を示す概要図である。

【図 5】

本実施形態におけるジョブ発行手段の処理の流れを示すフローチャート図である。

【図 6】

第 1 の本実施形態における「ジョブ発行通知」送信手段の処理の流れを示すフローチャートである。

【図 7】

第 1 の本実施形態における「ジョブ発行通知」受信手段の処理の流れを示すフローチャートである。

【図 8】

第 1 の本実施形態におけるジョブ実行指示手段の処理の流れを示すフローチャートである。

【図 9】

第 2 の本実施形態における「ジョブ発行通知」送信手段の処理の流れを示すフローチャートである。

【図 1 0】

第 2 の本実施形態における「ジョブ発行通知」受信手段の処理の流れを示すフローチャートである。

【図 1 1】

第 2 の本実施形態におけるジョブ実行指示手段の処理の流れを示すフローチャートである。

【図 1 2】

第 1、第 2 の本実施形態のジョブデータおよびジョブ実行指示コマンドデータで使用されるオペレーションを示す図である。

【図 1 3】

各オペレーションの詳細なデータ構造図である。

【図 1 4】

各オペレーションの詳細なデータ構造図である。

【図 1 5】

各オペレーションの詳細なデータ構造図である。

【図 1 6】

各オペレーションの詳細なデータ構造図である。

【図 1 7】

各オペレーションの詳細なデータ構造図である。

【図 1 8】

ジョブデータおよびジョブ実行指示コマンドデータで使用されるジョブ属性を示す図である。

【図 1 9】

ジョブ発行時のジョブデータの送信シーケンス図である。

【図 2 0】

ジョブ実行指示時のコマンドデータの送信シーケンス図である。

【図 2 1】

第 1 の実施形態におけるプリンタのブロック構成図である。

【図 2 2】

図 2 1 におけるジョブ管理テーブルの内容の一例を示す図である。

【図 2 3】

第 3 の実施形態におけるジョブ生成ソフトウェアおよびその周辺動作環境の概念図である。

【図 2 4】

第 3 の実施形態におけるジョブ生成ソフトウェアの動作関係を示す概要図である。

【図 2 5】

第 3 の本実施形態におけるジョブ発行手段の処理の流れを示すフローチャートである。

【図 2 6】

第 3 の本実施形態における「ジョブ発行通知」受信手段の処理の流れを示すフローチャートである。

【図 2 7】

第 3 の実施形態におけるジョブ実行指示手段の処理の流れを示すフローチャートである。

【図 2 8】

第 3 の実施形態におけるジョブ転送手段の処理の流れを示すフローチャートである。

【図 2 9】

第 3 の本実施形態のジョブデータおよびジョブ実行指示コマンドデータで 사용되는オペレーションを示す図である。

【図 3 0】

第 3 の実施形態で追加されたジョブ修正コマンドデータの構造図である。

【図 3 1】

第 3 の実施形態におけるジョブデータおよびジョブ実行指示コマンドデータで 사용되는ジョブ属性を示す図である。

【図 3 2】

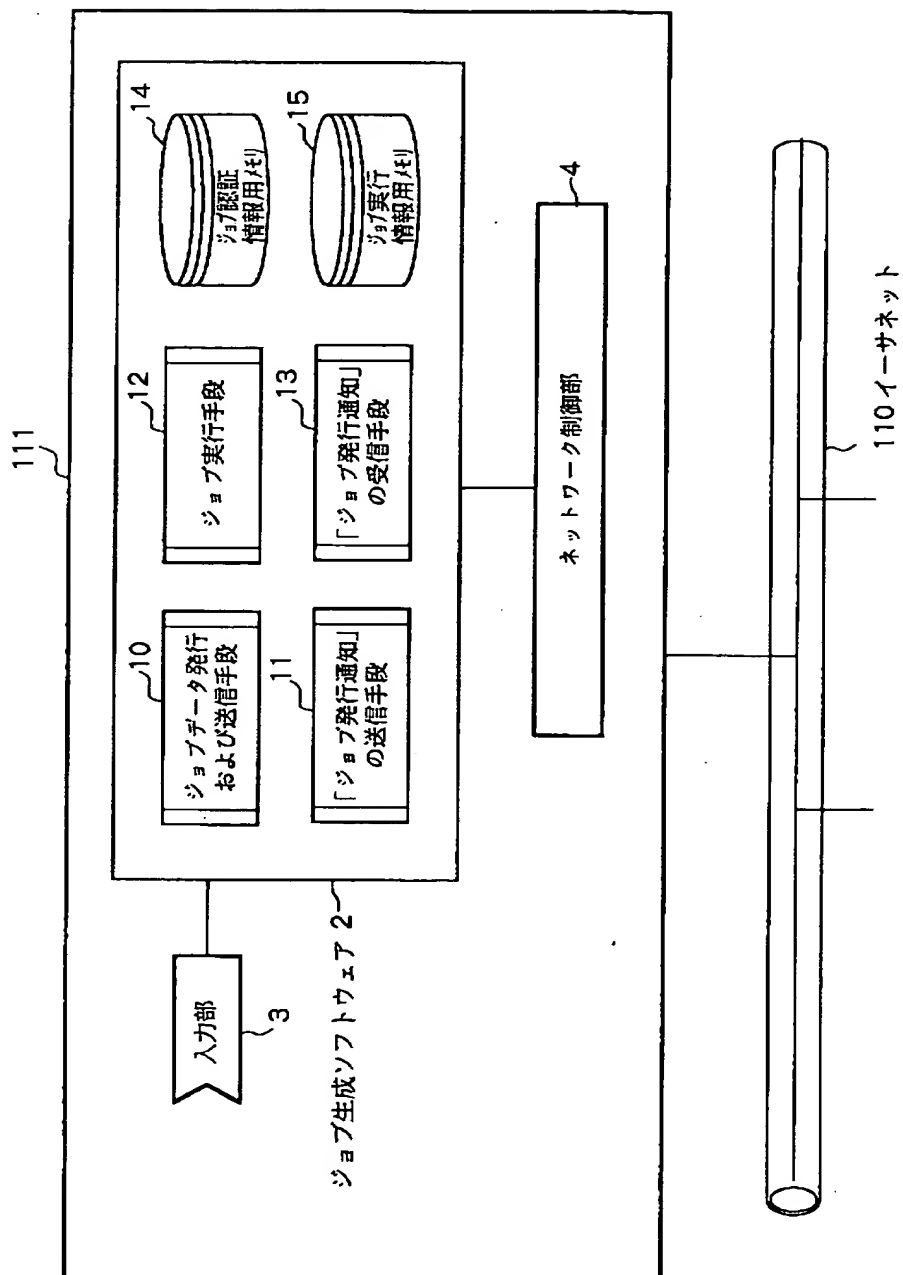
第 3 の実施形態におけるジョブ属性修正時のコマンドデータの送信シーケンスを示す図である。

【図 3 3】

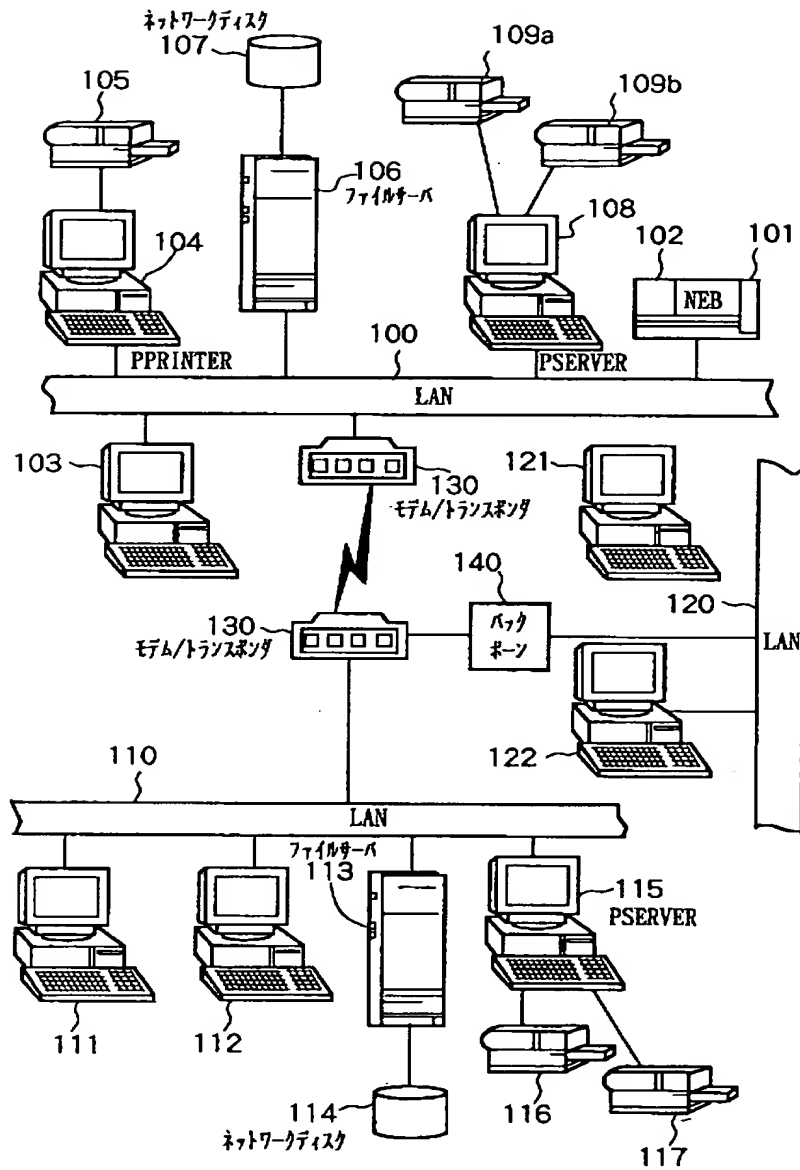
第 3 の実施形態におけるプリンタの動作処理手順を示すフローチャートである

【書類名】 図面

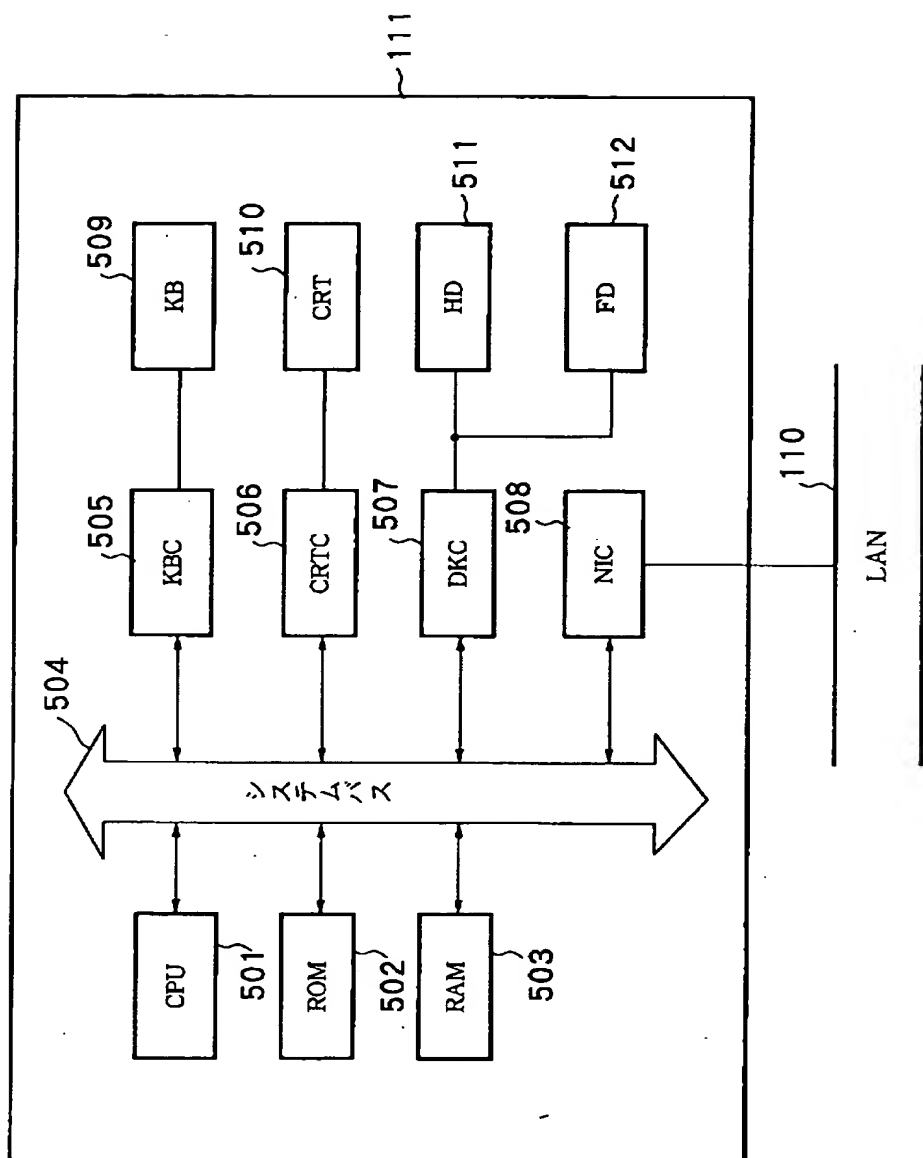
【図 1】



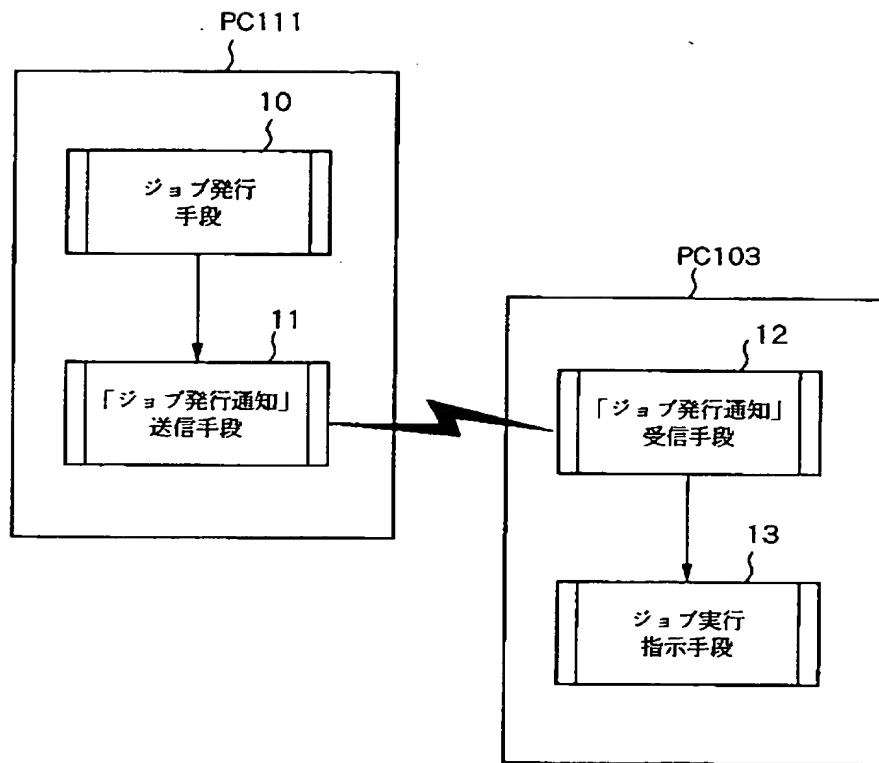
【図 2】



【図 3】

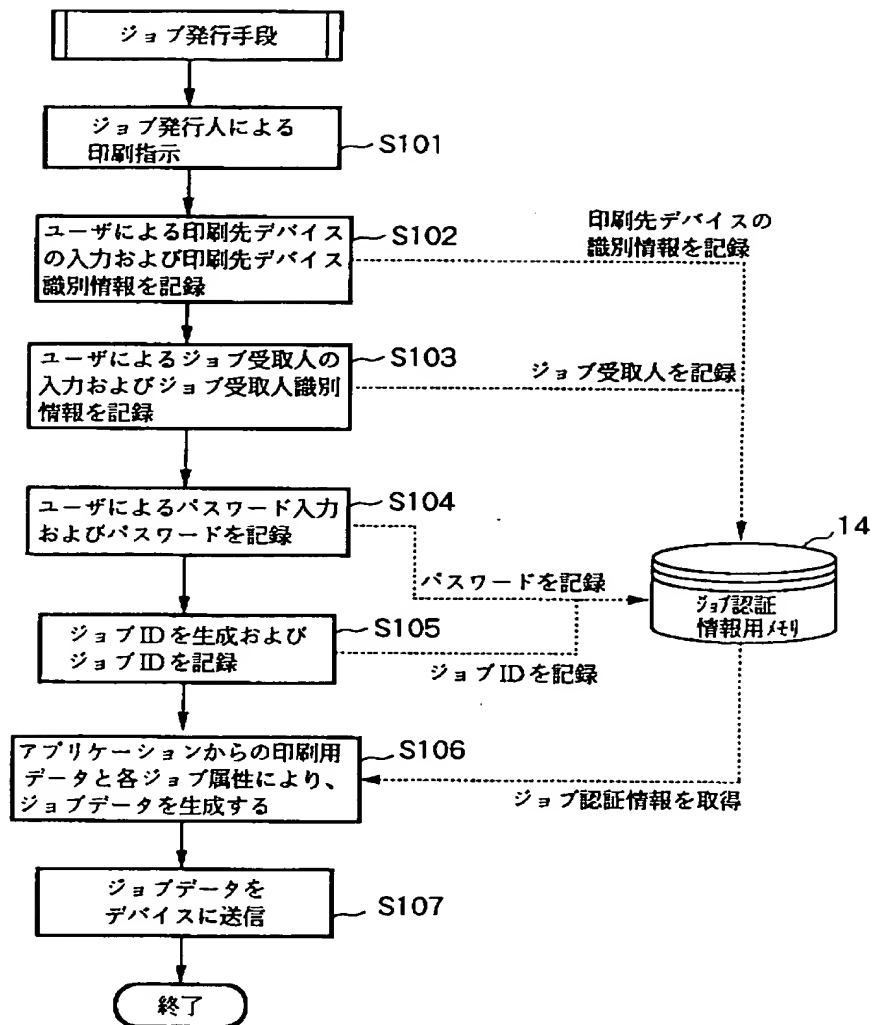


【図 4】

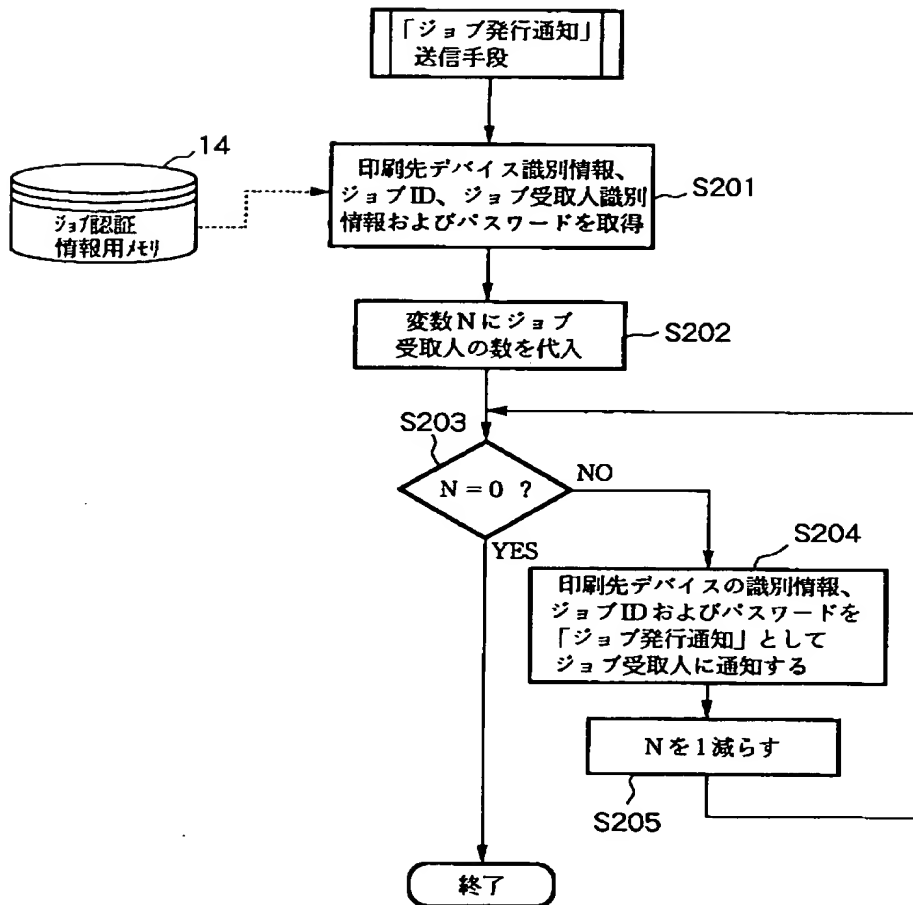




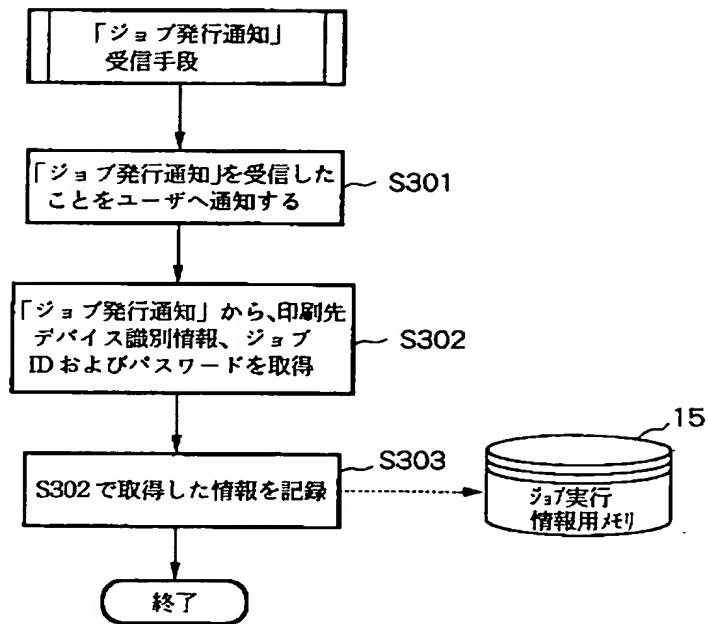
【図 5】



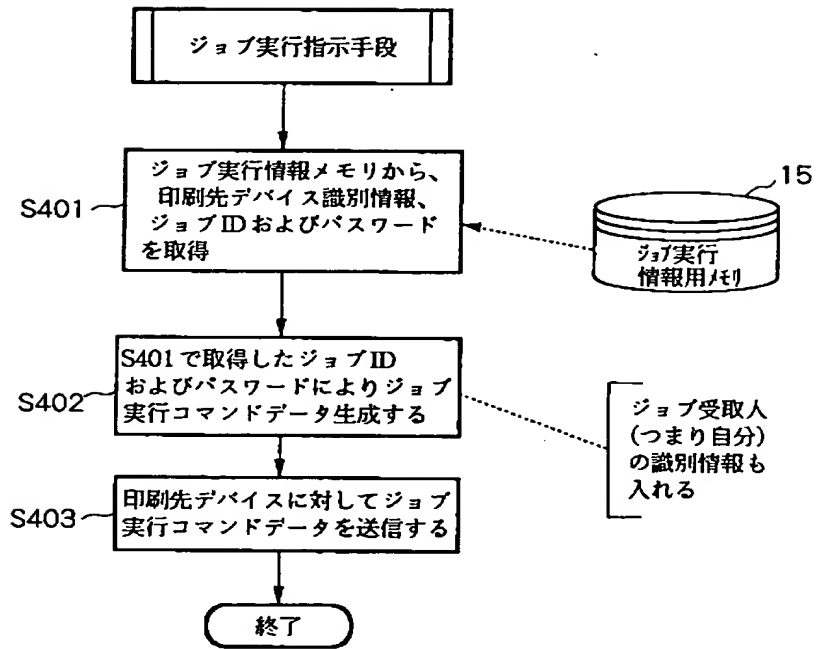
【図 6】



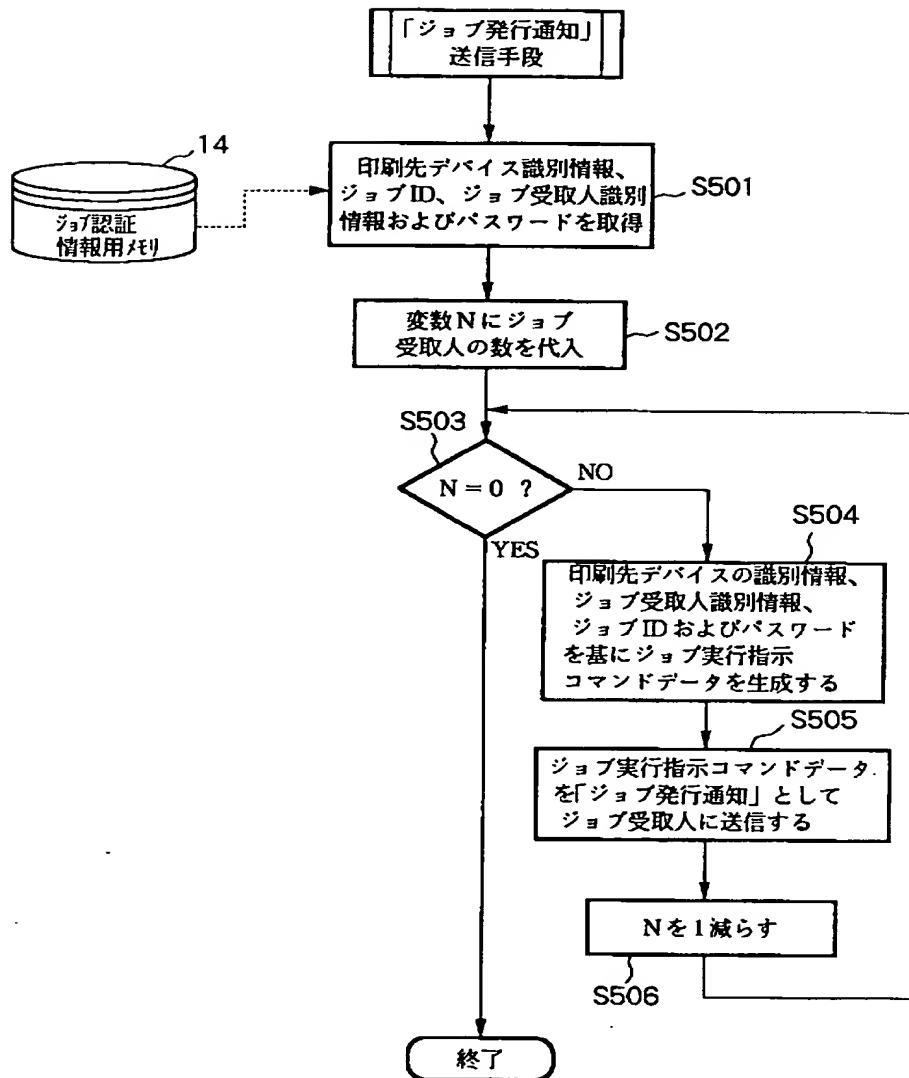
【図 7】



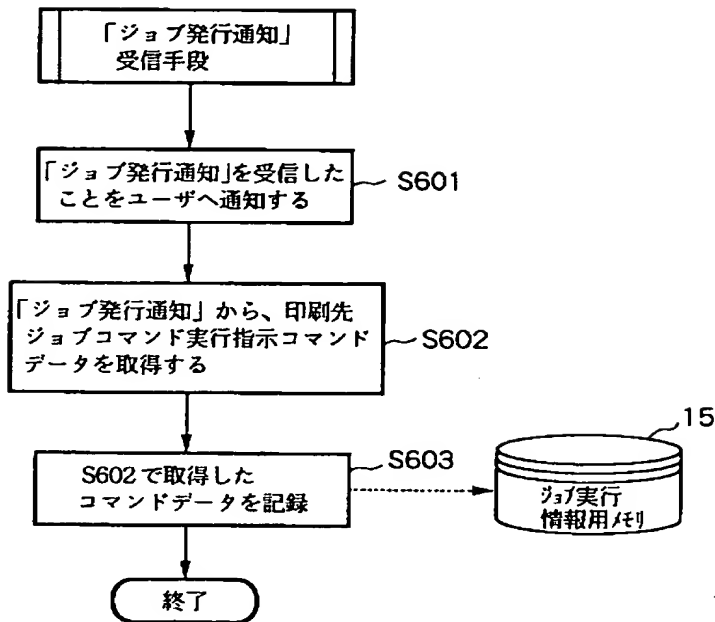
【図 8】



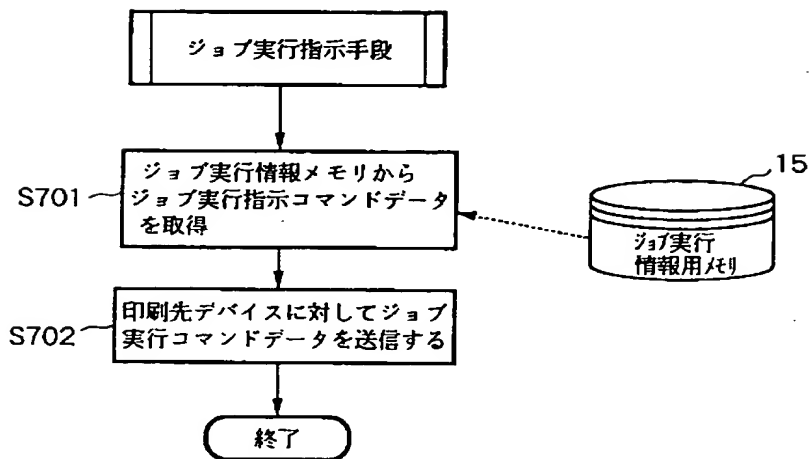
【図 9】



【図 10】



【図 11】



【図 1 2】

使用されるオペレーション

| オペレーション    | 機能概要     | スクリプト概要  |
|------------|----------|--|
| JobStart   | ジョブの発行開始 | SetJob でジョブ属性を設定<br>JobEnd で終了                         |
| SetJob     | ジョブ属性を設定 | ジョブ認証に必要なジョブ属性を<br>このオペレーションで設定                        |
| Send       | 印刷データを送信 |  |
| JobEnd     | ジョブの発行終了 |  |
| ExecuteJob | ジョブの実行指示 | ジョブ実行指示およびジョブ認証に必要な<br>ジョブ属性と共に、このオペレーションで<br>ジョブ実行を指示 |

【図 1 3】

JobStart

[送信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2             | 1            | 0           |
|-------------|---|---|---|---|---|---------------|--------------|-------------|
| 0~2         | Packet Header ID & Version                  |   |   |   |   |               |              |             |
| 3           | 0   | 0 | 0 | 0 | 0 | RqRep.<br>(1) | Cont.<br>(0) | Send<br>(0) |
| 4~5         | Operation Code = id_val_operation_job_start |   |   |   |   |               |              |             |
| 6~7         | Block Number = 0x0001~0xFFFF                |   |   |   |   |               |              |             |
| 8~9         | Parameter Length = 0x0009                   |   |   |   |   |               |              |             |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF             |   |   |   |   |               |              |             |
| 14~17       | Password = 0x00000000~0xFFFFFFFF            |   |   |   |   |               |              |             |
| 18~19       | Status Code = 0x0000                        |   |   |   |   |               |              |             |
| 20~23       | Object Handle (ジョブID)                       |   |   |   |   |               |              |             |

[返信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2             | 1              | 0            |
|-------------|---|---|---|---|---|---------------|----------------|--------------|
| 0~2         | Packet Header ID & Version                  |   |   |   |   |               |                |              |
| 3           | 0   | 0 | 0 | 0 | 0 | NoRep.<br>(0) | Cont.<br>(0/1) | Reply<br>(1) |
| 4~5         | Operation Code = id_val_operation_job_start |   |   |   |   |               |                |              |
| 6~7         | Block Number = 0x0001~0xFFFF                |   |   |   |   |               |                |              |
| 8~9         | Parameter Length = 0x0004                   |   |   |   |   |               |                |              |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF             |   |   |   |   |               |                |              |
| 14~17       | Password = 0x00000000~0xFFFFFFFF            |   |   |   |   |               |                |              |
| 18~19       | Status Code = 0x0000~0xFFFF                 |   |   |   |   |               |                |              |
| 20~23       | Object Handle (ジョブID)                       |   |   |   |   |               |                |              |



【図 1 4】

SetJob

[送信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2               | 1              | 0           |
|-------------|---|---|---|---|---|-----------------|----------------|-------------|
| 0~2         | Packet Header ID & Version                |   |   |   |   |                 |                |             |
| 3           | 0   | 0 | 0 | 0 | 0 | RqRep.<br>(0/1) | Cont.<br>(0/1) | Send<br>(0) |
| 4~5         | Operation Code = id_val_operation_set_job |   |   |   |   |                 |                |             |
| 6~7         | Block Number = 0x0001~0xFFFF              |   |   |   |   |                 |                |             |
| 8~9         | Parameter Length = 0x0003~0xFFFF          |   |   |   |   |                 |                |             |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF           |   |   |   |   |                 |                |             |
| 14~17       | Password = 0x00000000~0xFFFFFFFF          |   |   |   |   |                 |                |             |
| 18~19       | Status Code = 0x0000                      |   |   |   |   |                 |                |             |
| 20~21       | Attribute ID = 0x0000~0xFFFF              |   |   |   |   |                 |                |             |
| 22~         | Attribute Data = ANY                      |   |   |   |   |                 |                |             |

[返信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2             | 1              | 0            |
|-------------|---|---|---|---|---|---------------|----------------|--------------|
| 0~2         | Packet Header ID & Version                |   |   |   |   |               |                |              |
| 3           | 0   | 0 | 0 | 0 | 0 | NoRep.<br>(0) | Cont.<br>(0/1) | Reply<br>(1) |
| 4~5         | Operation Code = id_val_operation_set_job |   |   |   |   |               |                |              |
| 6~7         | Block Number = 0x0001~0xFFFF              |   |   |   |   |               |                |              |
| 8~9         | Parameter Length = 0x0003~0xFFFF          |   |   |   |   |               |                |              |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF           |   |   |   |   |               |                |              |
| 14~17       | Password = 0x00000000~0xFFFFFFFF          |   |   |   |   |               |                |              |
| 18~19       | Status Code = 0x0000~0xFFFF               |   |   |   |   |               |                |              |
| 20~21       | Attribute ID = 0x0000~0xFFFF              |   |   |   |   |               |                |              |
| 22~         | Attribute Data = ANY                      |   |   |   |   |               |                |              |

【図 1 5】

Send

[送信パケット]

| Bit<br>Byte | 7                                      | 6 | 5 | 4 | 3 | 2               | 1              | 0           |
|-------------|--|---|---|---|---|-----------------|----------------|-------------|
| 0~2         | Packet Header ID & Version             |   |   |   |   |                 |                |             |
| 3           | 0                                      | 0 | 0 | 0 | 0 | RqRep.<br>(0/1) | Cont.<br>(0/1) | Send<br>(0) |
| 4~5         | Operation Code = id_val_operation_send |   |   |   |   |                 |                |             |
| 6~7         | Block Number = 0x0000~0xFFFF           |   |   |   |   |                 |                |             |
| 8~9         | Parameter Length = 0x0002~0xFFFF       |   |   |   |   |                 |                |             |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF        |   |   |   |   |                 |                |             |
| 14~17       | Password = 0x00000000~0xFFFFFFFF       |   |   |   |   |                 |                |             |
| 18~19       | Status Code = 0x0000                   |   |   |   |   |                 |                |             |
| 20~21       | Send Mode = 0x00~0xFF                  |   |   |   |   |                 |                |             |
| 22~         | Send Source = ANY                      |   |   |   |   |                 |                |             |

[返信パケット]

| Bit<br>Byte | 7                                      | 6 | 5 | 4 | 3 | 2             | 1            | 0            |
|-------------|--|---|---|---|---|---------------|--------------|--------------|
| 0~2         | Packet Header ID & Version             |   |   |   |   |               |              |              |
| 3           | 0                                      | 0 | 0 | 0 | 0 | NoRep.<br>(0) | Cont.<br>(0) | Reply<br>(1) |
| 4~5         | Operation Code = id_val_operation_send |   |   |   |   |               |              |              |
| 6~7         | Block Number = 0x0000~0xFFFF           |   |   |   |   |               |              |              |
| 8~9         | Parameter Length = 0x0000              |   |   |   |   |               |              |              |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF        |   |   |   |   |               |              |              |
| 14~17       | Password = 0x00000000~0xFFFFFFFF       |   |   |   |   |               |              |              |
| 18~19       | Status Code = 0x0000~0xFFFF            |   |   |   |   |               |              |              |

【図 1 6】

JobEnd

[送信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2               | 1            | 0           |
|-------------|---|---|---|---|---|-----------------|--------------|-------------|
| 0~2         | Packet Header ID & Version                |   |   |   |   |                 |              |             |
| 3           | 0   | 0 | 0 | 0 | 0 | RqRep.<br>(0/1) | Cont.<br>(0) | Send<br>(0) |
| 4~5         | Operation Code = id_val_operation_job_end |   |   |   |   |                 |              |             |
| 6~7         | Block Number = 0x0001~0xFFFF              |   |   |   |   |                 |              |             |
| 8~9         | Parameter Length = 0x0001                 |   |   |   |   |                 |              |             |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF           |   |   |   |   |                 |              |             |
| 14~17       | Password = 0x00000000~0xFFFFFFFF          |   |   |   |   |                 |              |             |
| 18~19       | Status Code = 0x0000                      |   |   |   |   |                 |              |             |
| 20          | Control Flag = 0x0001                     |   |   |   |   |                 |              |             |

[返信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2             | 1              | 0            |
|-------------|---|---|---|---|---|---------------|----------------|--------------|
| 0~2         | Packet Header ID & Version                |   |   |   |   |               |                |              |
| 3           | 0   | 0 | 0 | 0 | 0 | NoRep.<br>(0) | Cont.<br>(0/1) | Reply<br>(1) |
| 4~5         | Operation Code = id_val_operation_job_end |   |   |   |   |               |                |              |
| 6~7         | Block Number = 0x0001~0xFFFF              |   |   |   |   |               |                |              |
| 8~9         | Parameter Length = 0x0005                 |   |   |   |   |               |                |              |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF           |   |   |   |   |               |                |              |
| 14~17       | Password = 0x00000000~0xFFFFFFFF          |   |   |   |   |               |                |              |
| 18~19       | Status Code = 0x0000~0xFFFF               |   |   |   |   |               |                |              |
| 20          | Control Flag = 0x00~0x01                  |   |   |   |   |               |                |              |
| 21~24       | Object Handle = 0x00000000~0xFFFFFFFF     |   |   |   |   |               |                |              |



【図 1 7】

## ExecuteJob

## [送信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2               | 1            | 0           |
|-------------|---|---|---|---|---|-----------------|--------------|-------------|
| 0~2         | Packet Header ID & Version                    |   |   |   |   |                 |              |             |
| 3           | 0   | 0 | 0 | 0 | 0 | RqRep.<br>(0/1) | Cont.<br>(0) | Send<br>(0) |
| 4~5         | Operation Code = id_val_operation_execute_job |   |   |   |   |                 |              |             |
| 6~7         | Block Number = 0x0001~0xFFFF                  |   |   |   |   |                 |              |             |
| 8~9         | Parameter Length = 0x0004                     |   |   |   |   |                 |              |             |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF               |   |   |   |   |                 |              |             |
| 14~17       | Password = 0x00000000~0xFFFFFFFF              |   |   |   |   |                 |              |             |
| 18~19       | Status Code = 0x0000                          |   |   |   |   |                 |              |             |
| 20~23       | Object Handle (ジョブID)                         |   |   |   |   |                 |              |             |
| 24~25       | Attribute ID = 0x0000~0xFFFF                  |   |   |   |   |                 |              |             |
| 26~         | Attribute Data = ANY                          |   |   |   |   |                 |              |             |

## [返信パケット]

| Bit<br>Byte | 7   | 6 | 5 | 4 | 3 | 2             | 1            | 0            |
|-------------|---|---|---|---|---|---------------|--------------|--------------|
| 0~2         | Packet Header ID & Version                    |   |   |   |   |               |              |              |
| 3           | 0   | 0 | 0 | 0 | 0 | NoRep.<br>(0) | Cont.<br>(0) | Reply<br>(1) |
| 4~5         | Operation Code = id_val_operation_execute_job |   |   |   |   |               |              |              |
| 6~7         | Block Number = 0x0001~0xFFFF                  |   |   |   |   |               |              |              |
| 8~9         | Parameter Length = 0x0008                     |   |   |   |   |               |              |              |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF               |   |   |   |   |               |              |              |
| 14~17       | Password = 0x00000000~0xFFFFFFFF              |   |   |   |   |               |              |              |
| 18~19       | Status Code = 0x0000~0xFFFF                   |   |   |   |   |               |              |              |
| 20~23       | Object Handle (ジョブID)                         |   |   |   |   |               |              |              |

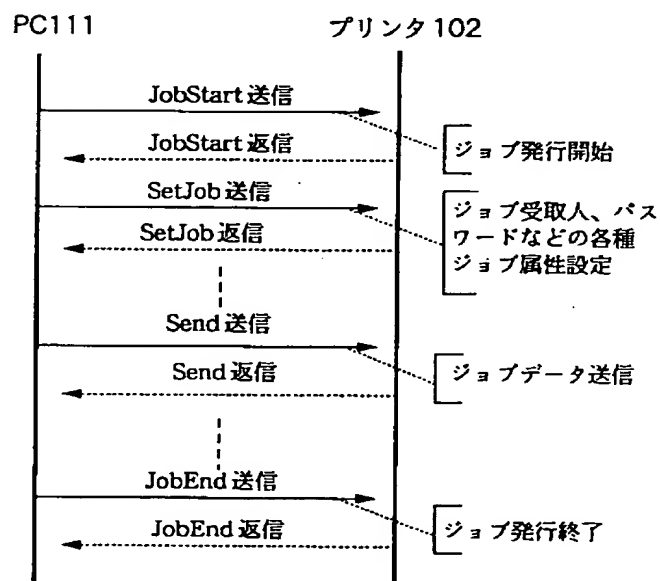
【図 1 8】

使用されるジョブ属性

| 分類    | 属性                 | 属性識別子                                |
|-------|--------------------|--------------------------------------|
| ジョブ属性 | ジョブ名               | id_att_job_name                      |
|       | クライアントの<br>ジョブ投入時刻 | id_att_job_submission_time_on_client |
|       | ジョブ発行人             | id_att_job_originator                |
|       | ジョブ受取人             | id_att_job_owner                     |
|       | ジョブ・パスワード          | id_att_job_password                  |

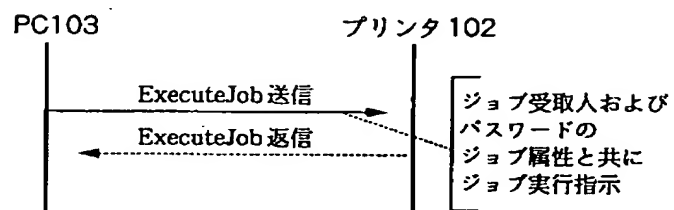
【図 1 9】

ジョブ発行時のジョブデータの送信シーケンス

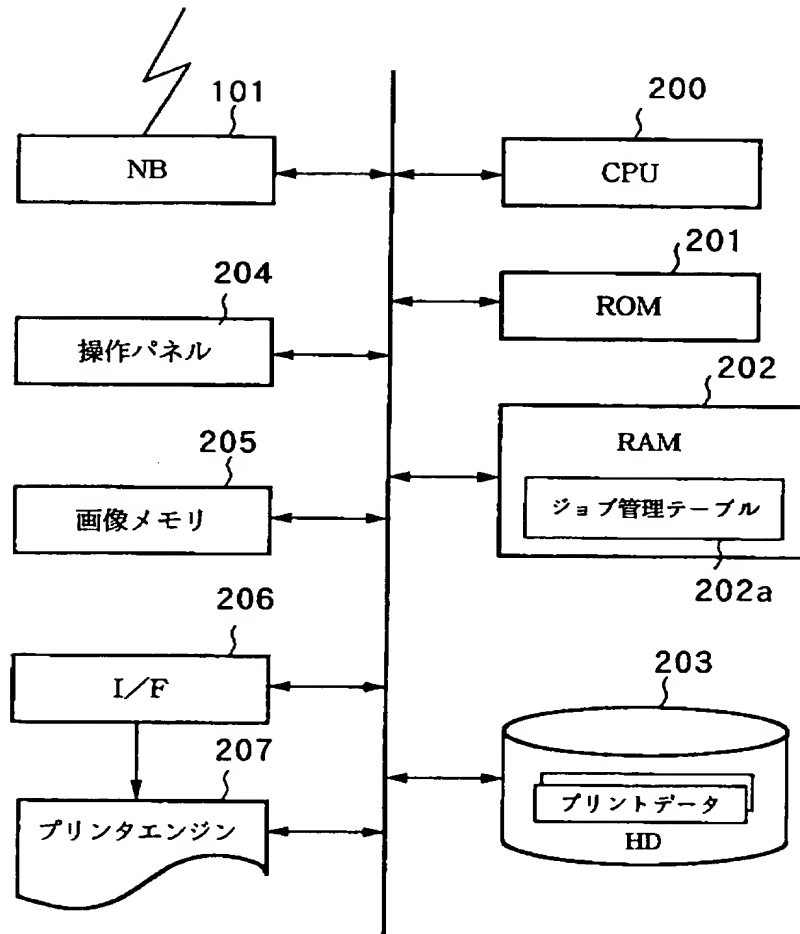


【図 2 0】

ジョブ実行指示時のコマンドデータの送信シーケンス



【図 2 1】



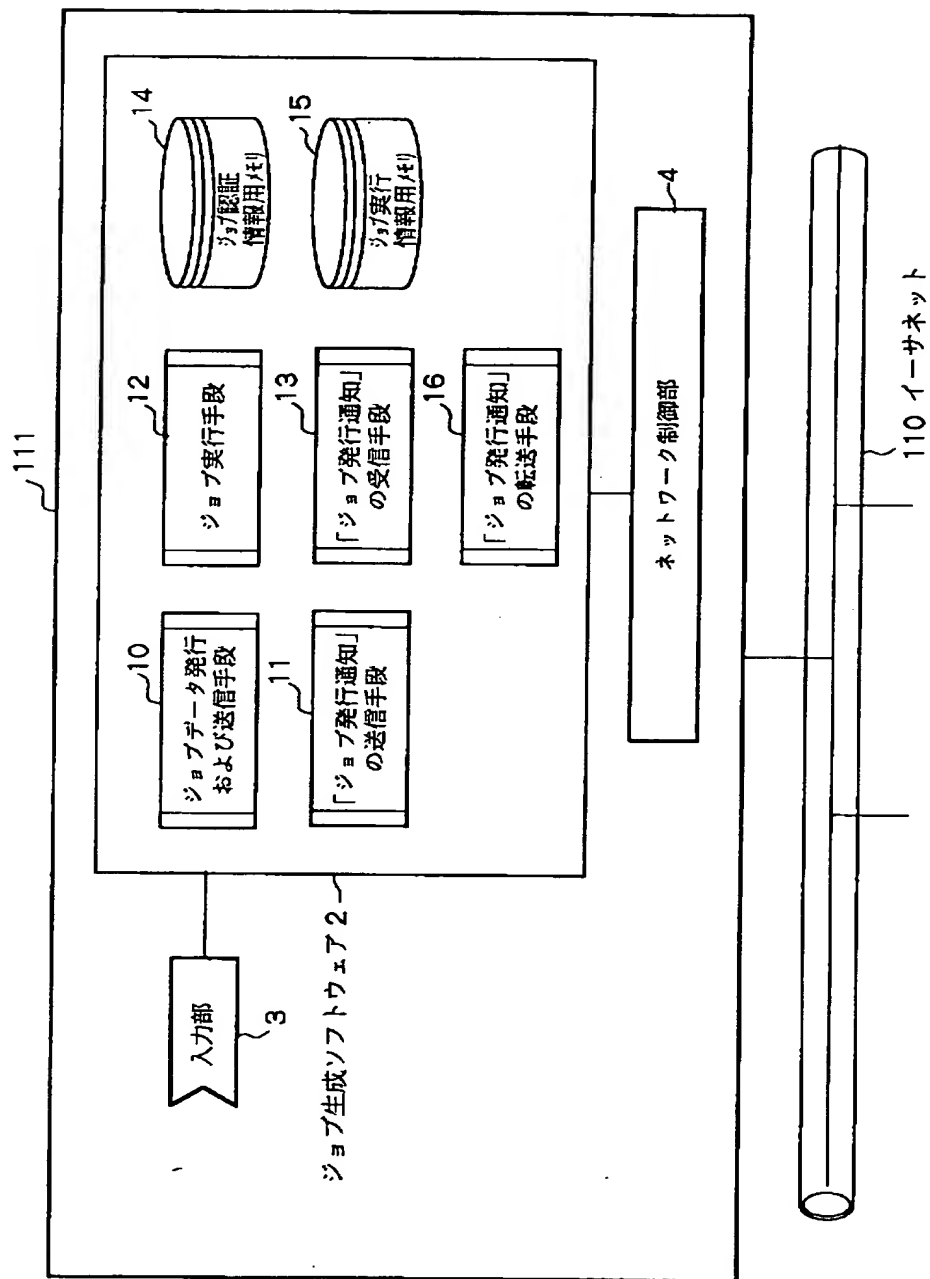
【図 2 2】

~ 202a

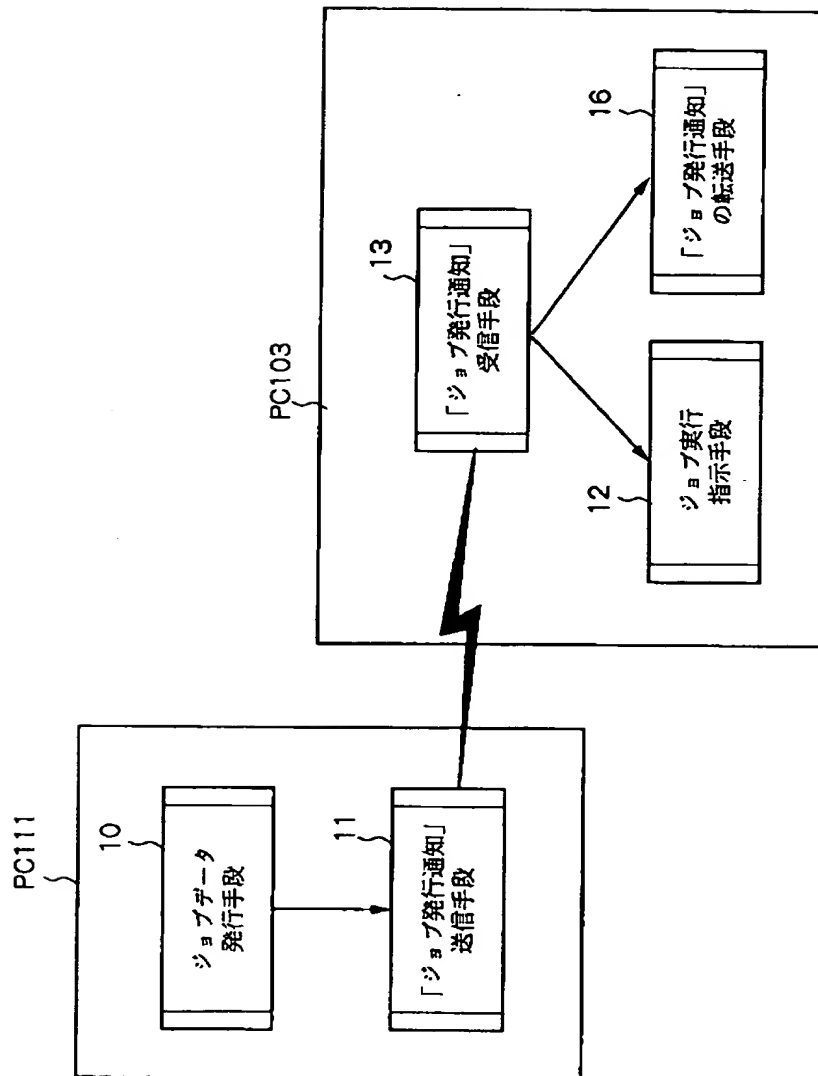
| ジョブID | 受取時間                   | 発行人   | パスワード | 受取人                          | 印刷データ     |
|-------|------------------------|-------|-------|------------------------------|-----------|
| 00011 | 2000/02/10<br>10:45:15 | ×××   | ##    | AAA(0),<br>BBB(0),<br>CCC(0) | 00011.PRN |
| - - - | - - -                  | - - - | - - - | - - -                        | - - -     |
| - - - | - - -                  | - - - | - - - | - - -                        | - - -     |



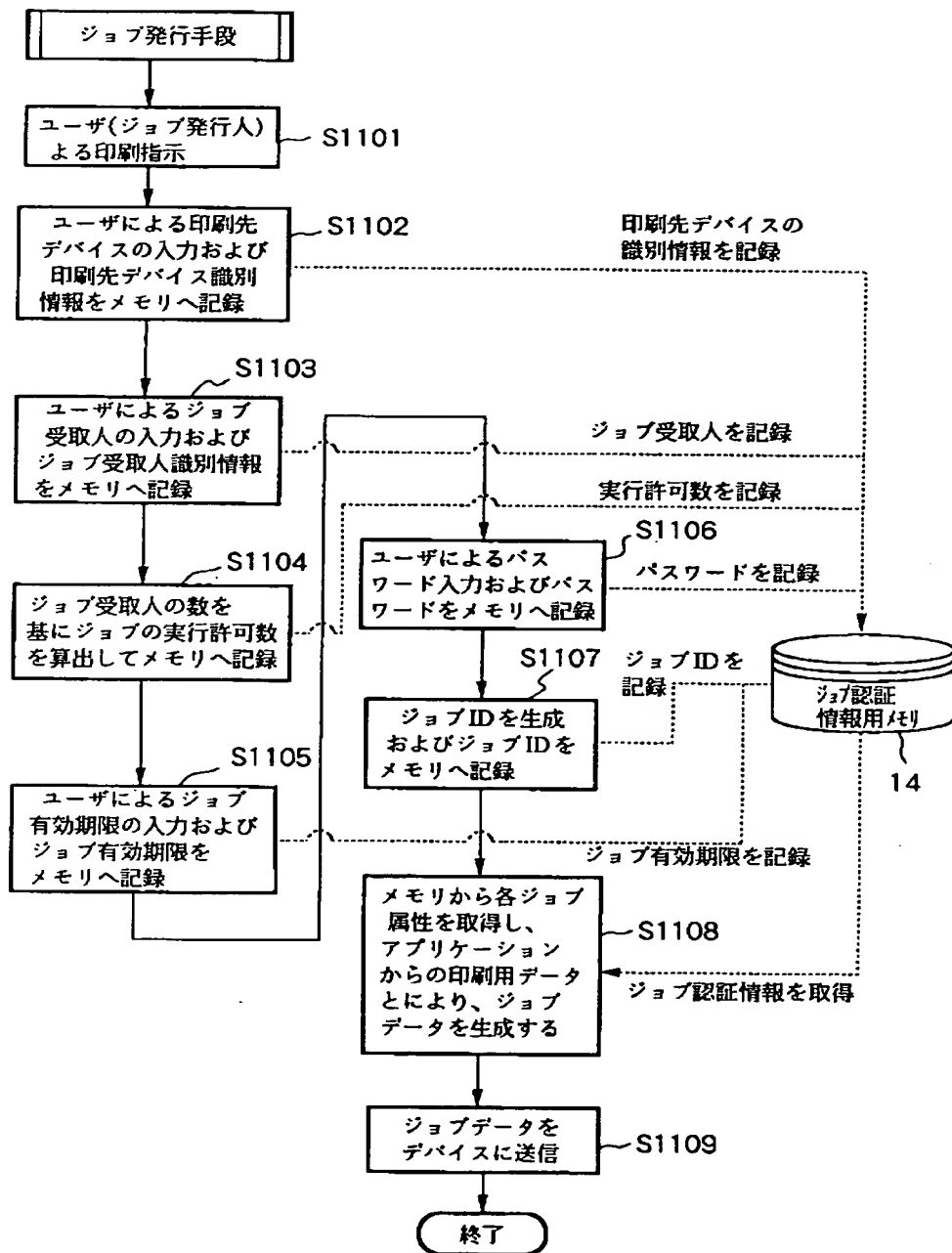
【図 23】



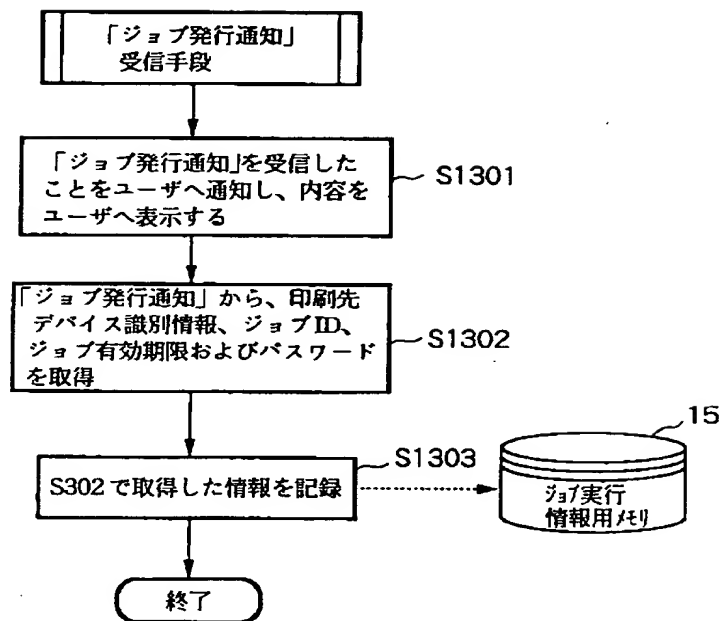
【図 2 4】



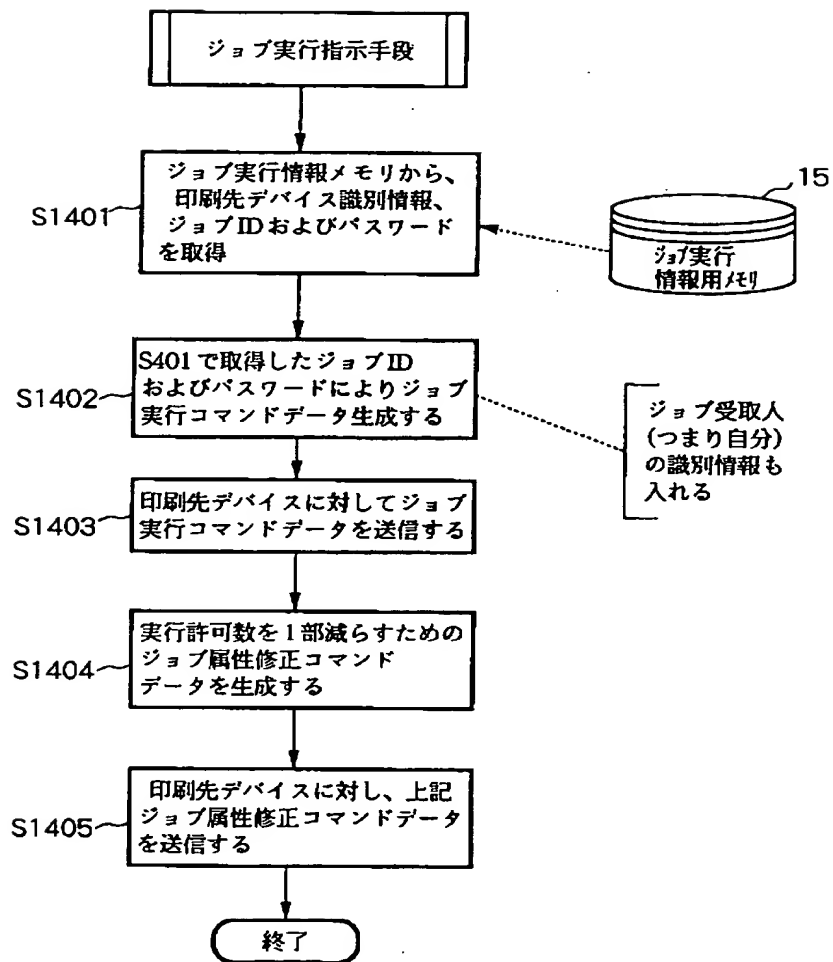
【図 25】



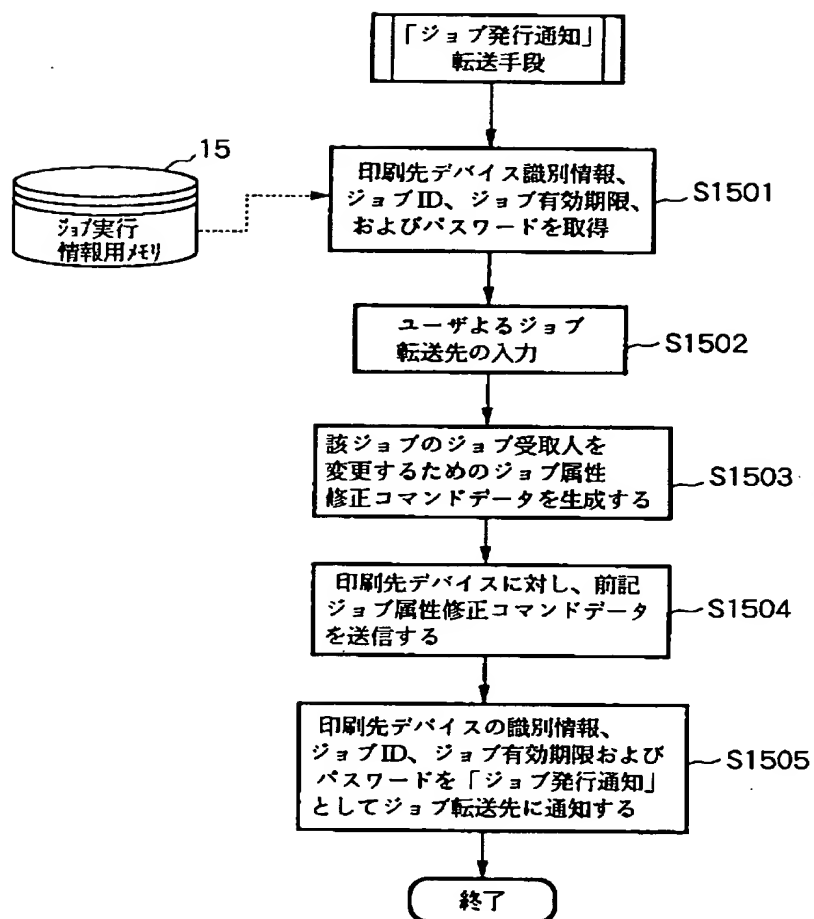
【図 2 6】



【図 2 7】



【図 2 8】



【図 29】

使用されるオペレーション

| オペレーション    | 機能概要     | スクリプト概要  |
|------------|----------|--|
| JobStart   | ジョブの発行開始 | SetJobでジョブ属性を設定<br>JobEndで終了                           |
| SetJob     | ジョブ属性を設定 | ジョブ認証に必要なジョブ属性を<br>このオペレーションで設定                        |
| Send       | 印刷データを送信 |  |
| JobEnd     | ジョブの発行終了 |  |
| Set        | ジョブ属性を修正 | すでに作成済みジョブデータ内の<br>ジョブ属性をこのオペレーションで修正                  |
| ExecuteJob | ジョブ実行指示  | ジョブ実行指示およびジョブ認証に必要な<br>ジョブ属性と共に、このオペレーションで<br>ジョブ実行を指示 |

【図 3 0】

Set

【送信パケット】

| Bit<br>Byte | 7                                     | 6 | 5 | 4 | 3 | 2               | 1              | 0           |
|-------------|---------------------------------------|---|---|---|---|-----------------|----------------|-------------|
| 0~2         | Packet Header ID & Version            |   |   |   |   |                 |                |             |
| 3           | 0                                     | 0 | 0 | 0 | 0 | RqRep.<br>(0/1) | Cont.<br>(0/1) | Send<br>(0) |
| 4~5         | Operation Code = id_val_operation_set |   |   |   |   |                 |                |             |
| 6~7         | Block Number = 0x0001~0xFFFF          |   |   |   |   |                 |                |             |
| 8~9         | Parameter Length = 0x0007~0xFFFF      |   |   |   |   |                 |                |             |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF       |   |   |   |   |                 |                |             |
| 14~17       | Password = 0x00000000~0xFFFFFFFF      |   |   |   |   |                 |                |             |
| 18~19       | Status Code = 0x0000                  |   |   |   |   |                 |                |             |
| 20~23       | Object Handle (ジョブID)                 |   |   |   |   |                 |                |             |
| 24~25       | Attribute ID = 0x0000~0xFFFF          |   |   |   |   |                 |                |             |
| 26~         | Attribute Data = ANY                  |   |   |   |   |                 |                |             |

【返信パケット】

| Bit<br>Byte | 7                                     | 6 | 5 | 4 | 3 | 2             | 1              | 0            |
|-------------|---------------------------------------|---|---|---|---|---------------|----------------|--------------|
| 0~2         | Packet Header ID & Version            |   |   |   |   |               |                |              |
| 3           | 0                                     | 0 | 0 | 0 | 0 | NoRep.<br>(0) | Cont.<br>(0/1) | Reply<br>(1) |
| 4~5         | Operation Code = id_val_operation_set |   |   |   |   |               |                |              |
| 6~7         | Block Number = 0x0001~0xFFFF          |   |   |   |   |               |                |              |
| 8~9         | Parameter Length = 0x0006~0xFFFF      |   |   |   |   |               |                |              |
| 10~13       | User ID = 0x00000000~0xFFFFFFFF       |   |   |   |   |               |                |              |
| 14~17       | Password = 0x00000000~0xFFFFFFFF      |   |   |   |   |               |                |              |
| 18~19       | Status Code = 0x0000~0xFFFF           |   |   |   |   |               |                |              |
| 20~23       | Object Handle (ジョブID)                 |   |   |   |   |               |                |              |
| 24~25       | Attribute ID = 0x0000~0xFFFF          |   |   |   |   |               |                |              |
| 26~         | Attribute Data = ANY                  |   |   |   |   |               |                |              |



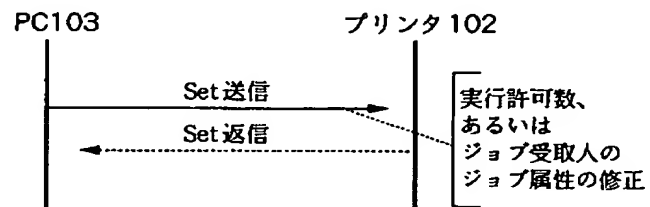
【図 3 1】

使用されるジョブ属性

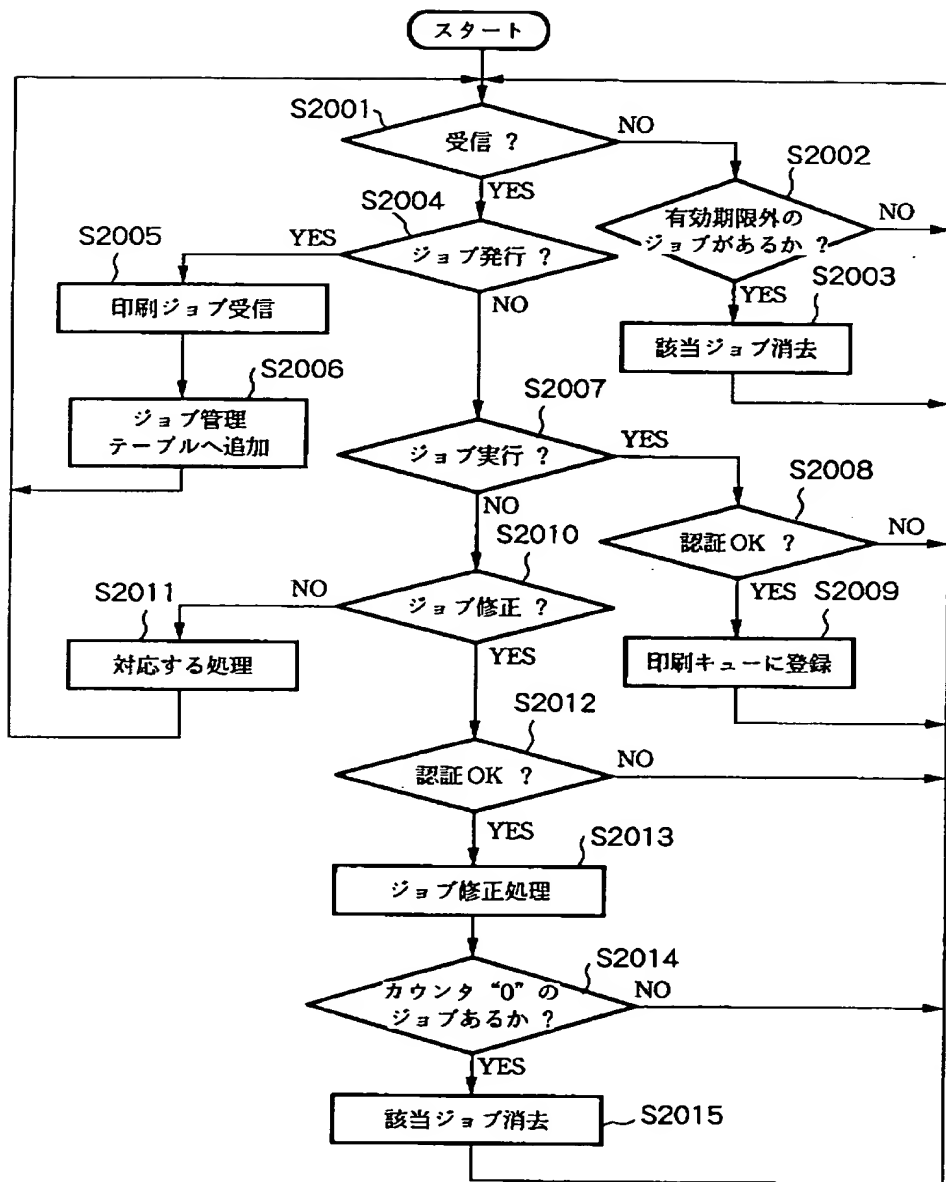
| 分類    | 属性                 | 属性識別子                                |
|-------|--------------------|--------------------------------------|
| ジョブ属性 | ジョブ名               | id_att_job_name                      |
|       | クライアントの<br>ジョブ投入時刻 | id_att_job_submission_time_on_client |
|       | ジョブ発行人             | id_att_job_originator                |
|       | ジョブ受取人             | id_att_job_owner                     |
|       | ジョブ有効期限            | id_att_job_discard_time              |
|       | ジョブ実行許可数           | is_att_job_abalable_count            |
|       | ジョブ・パスワード          | id_att_job_password                  |

【図 3 2】

ジョブ属性修正時のコマンドデータの送信シーケンス



【図 3 3】



【書類名】 要約書

【要約】

【課題】 ネットワーク上の第1のユーザが第2のユーザに対して情報を提供する際、その情報の元になる電子データを閲覧する機能を第2のユーザが使用する情報処理装置が無くても、当該第2のユーザが視覚可能に出力させる。

【解決手段】 ネットワーク110上のPC111で作成した情報を、ネットワーク100上のPC103のユーザに出力する際、PC111はプリンタ102に対して印刷ジョブを発行し、その発行した旨をPC103に通知する。PC103はこの通知を受け、プリンタ102に対してそのジョブの実行指示を与えることで初めて印刷ジョブが実行され、印刷物がPC103のユーザに渡される。

【選択図】 図2

出 願 人 履 歴 情 報

識別番号 [000001007]

1. 変更年月日 1990年 8月30日

[変更理由] 新規登録

住 所 東京都大田区下丸子3丁目30番2号

氏 名 キヤノン株式会社